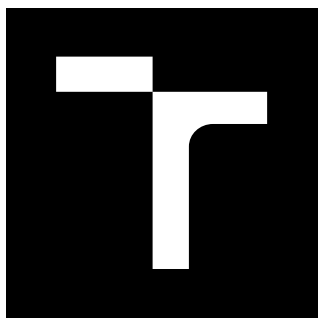


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA ELEKTROTECHNIKY**

**A KOMUNIKAČNÍCH TECHNOLOGIÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

**ÚSTAV TELEKOMUNIKACÍ**

DEPARTMENT OF TELECOMMUNICATIONS

**REKONSTRUKCE SIGNÁLU MODIFIKOVANÉHO  
EFEKTEM FADE-IN/FADE-OUT**

RECONSTRUCTION OF SIGNAL MODIFIED BY FADE-IN/FADE-OUT

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Petr Bača**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**doc. Mgr. Pavel Rajmic, Ph.D.**

**BRNO 2017**



# Bakalářská práce

bakalářský studijní obor **Audio inženýrství**  
Ústav telekomunikací

**Student:** Petr Bača

**ID:** 174440

**Ročník:** 3

**Akademický rok:** 2016/17

**NÁZEV TÉMATU:**

## Rekonstrukce signálu modifikovaného efektem fade-in/fade-out

### POKYNY PRO VYPRACOVÁNÍ:

Práce se bude věnovat úloze rekonstrukce původního signálu z jeho upravené verze, která vznikne náběhem (fade-in) nebo odběhem (fade-out) hlasitosti. V první variantě bude student předpokládat, že hlasitostní křivka je známa. Půjde tedy prakticky o úlohu dekvantizace audia. Student nejprve provede rešerši standardních metod pro dekvantizaci. Poté nastuduje základy problematiky tzv. řídkých reprezentací signálů jakožto odrazový můstek pro novou formulaci dekvantizační úlohy. Student tuto formulaci provede a navrhne algoritmus, který úlohu bude řešit. Student algoritmus implementuje v MATLABu a otestuje na simulovaných i reálných nahrávkách. Nakonec student podobně zpracuje druhou variantu úlohy, která bude navíc obsahovat odhad průběhu hlasitostní křivky (která v praxi obvykle není známa).

### DOPORUČENÁ LITERATURA:

[1] Hrbáček, R., Rajmic, P., Veselý, V., Špiřík, J. Řídké reprezentace signálů: úvod do problematiky, Elektrevue, 2011. ISSN 1213-1539

[2] Smékal, Z.: Analýza signálů a soustav. Skriptum. VUT v Brně, 2012.

**Termín zadání:** 1.2.2017

**Termín odevzdání:** 8.6.2017

**Vedoucí práce:** doc. Mgr. Pavel Rajmic, Ph.D.

**Konzultant:**

**doc. Ing. Jiří Mišurec, CSc.**  
*předseda oborové rady*

### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## ABSTRAKT

Bakalářská práce nese teoretický podklad pro zpracování speciální úlohy dekvantizace, a to rekonstrukce signálu postiženého efektem fade-in, fade-out. Je zde obsažen teoretický úvod shrnující základní poznatky o převodu z analogové oblasti do digitální. Dále zde nalézáme objasnění fenoménu řídkých reprezentací signálů, je zde formulována dekvantizační úloha a nadnesen algoritmus k jejímu výpočtu. Práce popisuje realizaci úlohy a přináší shrnutí výsledků celého procesu.

## KLÍČOVÁ SLOVA

Fade-in, fade-out, dekvantizace, rekonstrukce, řídké reprezentace, bitová hloubka.

## ABSTRACT

This thesis contains the theory needed to solve the special problem of bit-depth expansion. The goal is to reconstruct the signal which suffered from application of the fade-in, fade-out effect. The theory includes information of analog to digital conversion and the theory of sparse representations. Thesis formulates the task of bit-depth expansion and advises the algorithm to solve it. Furthermore, the realization of the issue is discussed and the results are given.

## KEYWORDS

Fade-in, fade-out, bit-depth expansion, reconstruction, sparse representations, bit-depth.

BAČA, Petr *Rekonstrukce signálu modifikovaného efektem fade-in, fade-out*: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2017. 48 s. Vedoucí práce byl doc. Mgr. Pavel Rajmic, PhD.

## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Rekonstrukce signálu modifikovaného efektem fade-in, fade-out“ jsem vypracoval(a) samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

podpis autora(-ky)

## PODĚKOVÁNÍ

Tímto bych chtěl poděkovat vedoucímu mé bakalářské práce, doc. Mgr. Pavlu Rajmici, Ph.D. za odborné vedení, čas strávený při konzultacích, ochotu a vstřícnost při domlouvání jejich termínů a trpělivost.

Brno .....

.....

podpis autora(-ky)

## PODĚKOVÁNÍ

Výzkum popsáný v této bakalářské práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Brno .....

.....  
podpis autora(-ky)

# OBSAH

<b>Úvod</b>	<b>12</b>
<b>1 Teoretický úvod</b>	<b>13</b>
1.1 AD/DA převod . . . . .	13
1.2 Vzorkování . . . . .	13
1.3 Kvantování . . . . .	14
1.4 D/D převod . . . . .	16
1.5 Důvody navyšování bitové hloubky a vzorkovací frekvence signálu . .	16
1.5.1 Důvody navýšení vzorkovací frekvence . . . . .	16
1.5.2 Důvody navýšení bitové hloubky . . . . .	17
<b>2 Způsob navýšení bitové hloubky</b>	<b>19</b>
2.1 Zero padding . . . . .	19
2.2 Praxe při úpravě signálu . . . . .	20
<b>3 Řídké reprezentace signálů</b>	<b>22</b>
3.1 Představení řídké reprezentace signálů . . . . .	22
3.1.1 Značení . . . . .	22
3.1.2 Pojmy . . . . .	23
3.2 Aditivní model signálu . . . . .	23
3.3 Řídké řešení . . . . .	24
3.4 Strukturovaná řídkost . . . . .	25
3.4.1 Strukturovaná řídkost v audio signálech . . . . .	25
<b>4 Signál modifikovaný efektem fade-out, resp. fade-in</b>	<b>27</b>
4.1 Simulace řešení úlohy v analogové oblasti . . . . .	27
4.2 Standardní řešení zadání . . . . .	27
4.3 Řešení za využití řídkosti signálu . . . . .	29
4.3.1 Formulace úlohy pro konstantní zvýšení bitové hloubky . . . .	29
4.3.2 Formulace úlohy pro signál zatížený efektem fade-out . . . .	30
4.4 Proximální operátor . . . . .	31
4.5 Navržení algoritmu pro řešení úlohy . . . . .	31
4.5.1 Formulace úlohy do vzoru proximálního algoritmu . . . . .	31
4.5.2 Volba algoritmu . . . . .	32
4.5.3 Frame, těsný frame, Gaborův frame . . . . .	32
4.5.4 Algoritmus Douglas-Rachford . . . . .	32



<b>5</b>	<b>Programové řešení - simulovaný a reálný signál</b>	<b>35</b>
5.1	Programové řešení - simulovaný signál . . . . .	35
5.1.1	Výchozí stav . . . . .	35
5.1.2	Základní parametry signálu . . . . .	35
5.1.3	Slovník a vytvoření simulovaného signálu . . . . .	35
5.1.4	Kvantizace signálu . . . . .	35
5.1.5	Vykreslení dosavadních výsledků . . . . .	35
5.1.6	Dekvantizace za využití řídkosti . . . . .	36
5.1.7	Vykreslení výsledků . . . . .	36
5.2	Programové řešení - reálný signál . . . . .	36
5.2.1	Nahrání a analýza signálu, zobrazení spektra . . . . .	36
5.2.2	Definice obálky . . . . .	36
5.2.3	Aplikace obálky . . . . .	37
5.2.4	Určení kvantizačních hladin, rozhodovacích úrovní a kvantizace ztlumeného signálu . . . . .	37
5.2.5	Pouhé pronásobení signálu inverzní obálkou . . . . .	39
5.2.6	Definice těsného framu . . . . .	39
5.2.7	Dekvantizace za využití řídkosti - rekonstrukce efektu fade-out . . . . .	39
5.2.8	Analýza . . . . .	39
5.2.9	Vykreslení výsledků . . . . .	39
5.2.10	Parametry ovlivňující výsledek dekvantizace . . . . .	40
<b>6</b>	<b>Výsledky bakalářské práce</b>	<b>41</b>
6.1	Kritéria hodnocení, testovací vzorky audia . . . . .	41
6.2	Spektra jednotlivých vzorků . . . . .	41
6.3	Signal-to-Noise ratio . . . . .	41
6.3.1	SNR při rekonstrukci z různých bitových hloubek pronásobením a řídkou dekvantizací . . . . .	41
6.3.2	SNR různých délek okna u vzorků kvantovaných deseti bity . . . . .	44
<b>7</b>	<b>Závěr</b>	<b>45</b>
	<b>Literatura</b>	<b>46</b>
<b>8</b>	<b>Obsah příloženého CD</b>	<b>48</b>

# SEZNAM OBRÁZKŮ

1.1	Porovnání spojitého signálu s diskretním, tedy výsledek vzorkování. Inspirováno zdrojem [2, strana 6]. . . . .	14
1.2	Kvantování vzorků signálu, inspirováno zdrojem [5, strana 23]. . . . .	15
2.1	Výsledek dekvantizace signálu bez získání nové informace – nové kvantizační hladiny zůstávají volné. Konkrétní případ dekvantizace ze 3 na 5 bitů. . . . .	19
4.1	Aplikace efektu fade-out. Příklad zobrazuje známou obálku. . . . .	27
4.2	Znázornění kompenzace efektu zesílením. . . . .	28
5.1	Názorný příklad řídkého spektra testovaného signálu . . . . .	37
5.2	Znázornění vlivu efektu fade-out na původní signál . . . . .	38
5.3	Velikost změny na jednotlivých vzorcích, ke které došlo při simulaci kvantizace . . . . .	38
6.1	Pop . . . . .	42
6.2	Folk . . . . .	42
6.3	Řeč . . . . .	42
6.4	Kytara . . . . .	42
6.5	Pouhé pronásobení . . . . .	43
6.6	Řídká dekvantizace . . . . .	43

# SEZNAM TABULEK

1.1	Přehled závislosti bitové hloubky převodníku na počtu jeho kvanti- začních hladin. . . . .	15
1.2	Přehled použití různých velikostí vzorkovací frekvence. . . . .	17
1.3	Závislost dynamiky signálu na bitové hloubce převodníku a typické možnosti jejího využití. Čerpáme ze zdrojů [7] a [8]. . . . .	17
6.1	Signal-to-noise ratio rekonstrukce z různých bitových hloubek pomocí řádké DEQ a prostým násobením . . . . .	43

## SEZNAM VÝPISŮ

4.1	Původní zápis funkce f2.prox v prostředí Matlab. . . . .	33
4.2	Nový zápis funkce f2.prox v prostředí Matlab. . . . .	34

# ÚVOD

## Zadání práce

Zadání práce, tedy rekonstrukce signálu postprodukčně poznamenaného náběhem a odběhem hlasitosti, vede k úloze o dekvantizaci. Tu se snažíme optimalizovat za využití vlastnosti řídkosti signálu. Jak bude níže osvětleno, standardní dekvantizační metody nepřinášejí pro zvukovou praxi dostatečně uspokojivé výsledky, proto se snažíme zavést do této problematiky fenomén řídkosti signálu.

## Orientace v textu

V teoretickém úvodu budou osvětleny základní operace, které jsou podstatné pro zpracování dané problematiky. Následuje podrobnější kapitola o problematice kvantizace a dekvantizace signálu, kterou se snažíme touto prací optimalizovat pro konkrétní zadání. Poté rozebereme samotný fenomén řídké reprezentace signálů. Čtvrtá kapitola přistupuje již k vlastní implementaci řídkosti na naši konkrétní úlohu. Najdeme zde kompletní teoretický předpoklad pro realizaci, která je předmětem následujících kapitol, včetně představení proximálních algoritmů a návržení konkrétního vhodného algoritmu. Pátá kapitola popisuje kompletní programové řešení a vyznačuje nejdůležitější prvky, které hrají roli v celé realizaci úlohy. V šesté úloze jsou shrnuty výsledky a vyneseno hodnocení.

# 1 TEORETICKÝ ÚVOD

## 1.1 AD/DA převod

V soudobé audio elektrotechnice je analogově digitální a digitálně analogový (dále A/D, D/A) převod podstatný pojem. Je to operace nezbytná při jakékoli digitální úpravě zvuku. Kvalita procesu převodu analogového signálu na digitální je stěžejní pro zachování původních informací obsažených v analogovém signálu a pro další práci se signálem již na rozhraní digitálním. Tento převod zahrnuje i vzorkování a kvantizaci signálu.

## 1.2 Vzorkování

Vzorkováním [1], [2], [3] signálu rozumíme jeho převod ze spojitě domény do diskrétní. Jelikož digitální zařízení nedokáže pracovat s plynulým signálem, potřebujeme signál rozdělit na vzorky. A/D a D/A převodníky se primárně liší vzorkovací frekvencí. Vzorkovací frekvence udává, kolik vzorků za sekundu vytvoříme, tedy kolik jednotlivých impulsů signálu za sekundu získáme. Náhled převodu ukazuje obrázek 1.1.

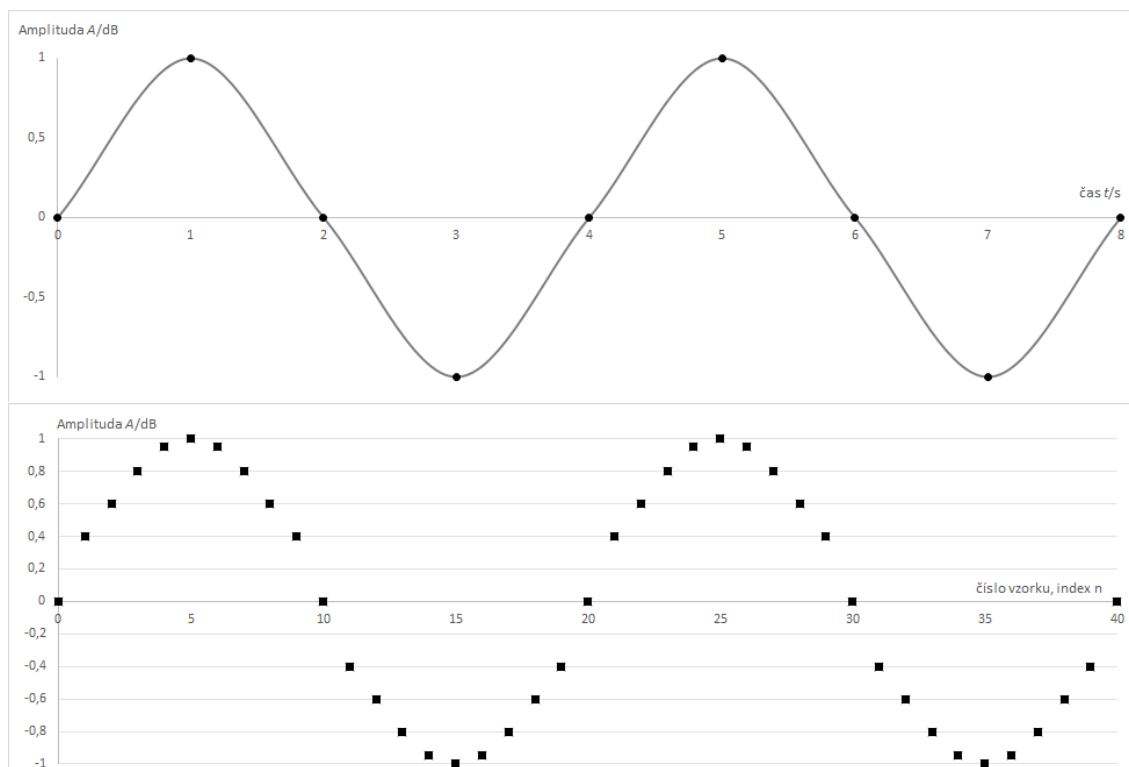
Při vzorkování jakéhokoli, tedy i audio signálu je třeba dodržet Shannon-Nyquistův teorém, tzv. vzorkovací poučku, kdy musí platit, že:

$$f_{vz} > 2 \cdot f_{max}, \quad (1.1)$$

kde  $f_{vz}$  zastupuje vzorkovací frekvenci a  $f_{max}$  nejvyšší vzorkovanou frekvenci.

Vzorkujeme-li audiosignál, nemá smysl vzorkovat frekvence vyšší než 20 kHz. I kdybychom vzorkovali signál výše, nedochází k žádnému zlepšení kvality. Průměrné zdravé ucho dospělého člověka je totiž schopno zaznamenat nejvýše frekvenci 18 kHz. Existují však případy, kdy je i 18 kHz zbytečně vysoká, pracujeme totiž s omezenými audio signály. Například se můžeme setkat s telefonním audio signálem, který se pohybuje v rozmezí 200 Hz – 3,5 kHz [3].

Pokud není tato poučka zachována, dochází k aliasingu. Jde o chybu, která má závažný dopad na výsledný signál. Při vzorkování signálu se totiž vzorky zrcadlí vzhledem k polovině vzorkovací frekvence. Pokud je příliš nízká, vzorkovaný signál se zrcadlí zpět do prostoru již obsazeného předešlými vzorky, kde je ovlivní součtem. Z toho vyplývá, že aliasingu zabráníme, pokud do převáděného signálu nepropustíme vzorky o frekvenci vyšší, než je dvojnásobek frekvence vzorkovací. Tohoto je možné docílit zařazením antialiasingových filtrů typu dolní propust.

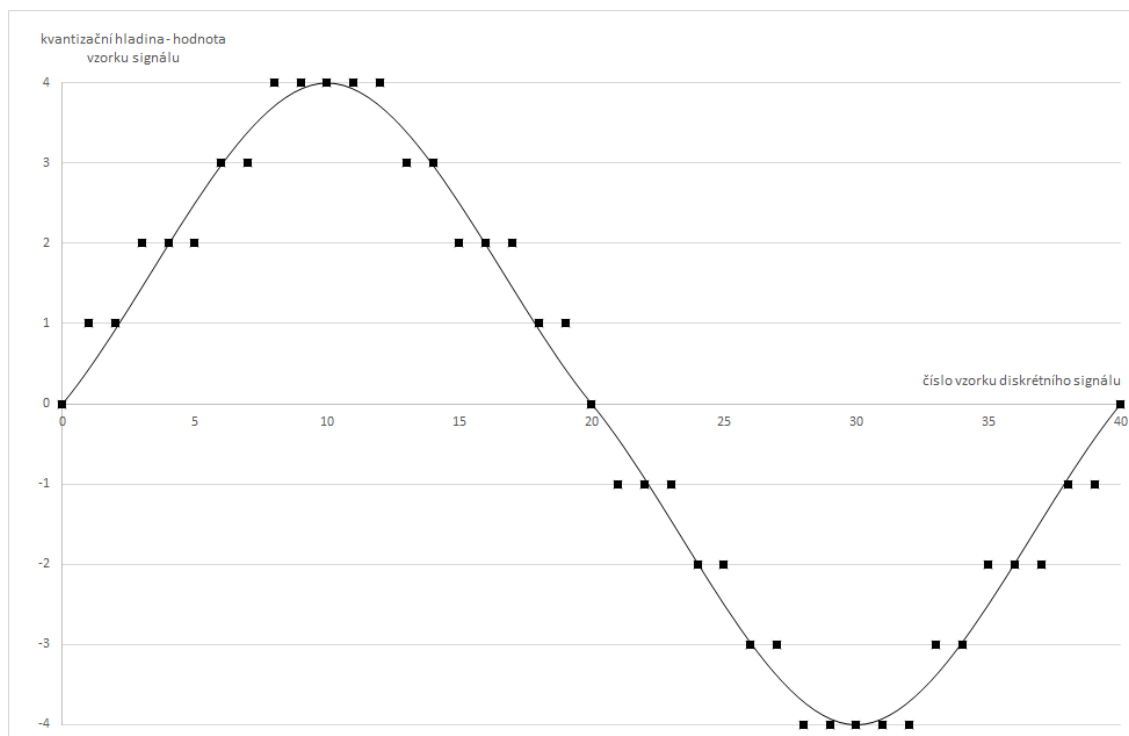


Obr. 1.1: Porovnání spojitého signálu s diskrétním, tedy výsledek vzorkování. Inspirováno zdrojem [2, strana 6].

### 1.3 Kvantování

Kvantování [2], [3] signálu znamená přiřazení každému jeho vzorku tzv. kvantizační hladinu. Analogový signál si můžeme představit jako signál s nekonečným množstvím kvantizačních hladin, kdy kvantizační krok, tedy vzdálenost mezi těmito hladinami, je nekonečně malý. V reálné práci digitálních technologií je však takováto rozlišovací schopnost nejen nezpracovatelná, ale především zbytečná. Přesto kvantováním vždy rozumíme proces degradační, jelikož snižujeme rozlišovací schopnost. Jemnost kvantování určuje bitová hloubka AD převodníku, pro audio je standard 16 bitů (CD kvalita), při náběru se záměrem další postprodukce se běžně zaznamenává 24, resp. 32 bitů [4]. Samotný proces kvantování pak znamená zaokrouhlení aktuální hodnoty signálu při pořízení vzorku obvykle na nejbližší kvantizační hladinu. Chyba oproti originální úrovni signálu se ve výsledku projeví coby kvantizační šum. Střední hodnotě mezi dvěma kvantizačními úrovněmi pak říkáme rozhodovací úroveň. Tuto metodu nazýváme pulsně-kódová modulace (PCM). Názorné zobrazení celého procesu je ukázáno na obrázku 1.2.

Je tedy dána omezená bitová hloubka signálu, chceme zaplnit co největší část z ní. Pro audiosignál, který je složený ze součtu funkcí s průběhem sinus, vždy



Obr. 1.2: Kvantování vzorků signálu, inspirováno zdrojem [5, strana 23].

volíme střed bitové hloubky na úrovni, kde amplituda signálu nabývá hodnoty nula. Audiosignál má výkon symetricky rozložený nad i pod touto hodnotou, proto touto volbou získáme nejlepší využití bitové hloubky. Počet bitů převodníku na vzorek exponenciálně ovlivní jeho bitovou hloubku. Stručný přehled vidíme v tabulce 1.1.

Bitová hloubka	Počet kvantizačních hladin
4	16
8	256
16	65 536
24	16 777 216
32	4 294 967 296
48	281 474 976 710 656
64	18 446 744 073 709 600 000

Tab. 1.1: Přehled závislosti bitové hloubky převodníku na počtu jeho kvantizačních hladin.



## 1.4 D/D převod

Výše byly doposud popsány pouze případy, kdy byl analogový signál převáděn na digitální. Proces kvantizace však probíhá i při úpravě bitové hloubky již jednou nakvantovaného signálu. Je to proces snadno přirovnatelný k A/D převodu, kdy opět proběhne navzorkování a hodnota jednotlivých vzorků se zaokrouhlí k nejbližším kvantizačním hladinám. Akci nazýváme rekvantizací, resp. snížením bitové hloubky signálu. Provádíme ji například pro úsporu místa na úložišti nebo pro dosažení nižších nároků na výpočetní kapacitu. Dalším důvodem pro rekvantizaci je dosažení kompatibility dvou signálů při práci v postprodukčním software, Digital Audio Workstation (dále jen DAW). Provádíme ji v případě, kdy dostačuje nižší bitová hloubka signálu. Viz [5], [6].

## 1.5 Důvody navyšování bitové hloubky a vzorkovací frekvence signálu

Obecně se dá shrnout, že větší vzorkovací frekvence a bitová hloubka přinese vyšší kvalitu výsledného signálu. Je třeba znát požadovaný výsledek, abychom byli pro svou práci schopni zvolit správný kompromis, například vzhledem k výpočetní náročnosti či finanční úspoře.

### 1.5.1 Důvody navýšení vzorkovací frekvence

Jak bylo uvedeno výše, zásadním nebezpečím při vzorkování signálu je vznik aliasingu. Dochází k němu, pokud vzorkujeme signál s frekvencí nižší než dvojnásobek nejvyšší vzorkované frekvence. Tomuto nežádoucímu jevu předejdeme, pokud před samotným navzorkováním zařadíme filtr typu dolní propust s mezním kmitočtem rovným polovině vzorkovacího kmitočtu převodníku. Tyto filtry však v praxi mají konečnou strmost, takže nejsou schopny signál filtrovat skokově. Nejlepším řešením tedy je zvolit mezní kmitočet filtru i vzorkovací frekvenci převodníku s rezervou. Tedy v případě vzorkovací frekvence volíme nikoli rovných 40 kHz, ale 44,1, resp. 48 kHz. Proto filtr nastavujeme dostatečně vysoko, aby jeho převodní charakteristika neovlivnila slyšitelné spektrum signálu, ale zároveň dostatečně nízko, aby nepropustil více než polovinu vzorkovací frekvence převodníku. Další navýšení vzorkovací frekvence snižuje nároky na strmost (řád) filtru, což se přímo projeví na jeho ceně. Dnes se setkáme s vzorkovacími frekvencemi násobně vyššími, např. 96, popř. 192 kHz. Druhým důvodem pro navýšení vzorkovací frekvence je umožnění většího postprodukčního zásahu do nahrávky. Audiosignál vzorkovaný s  $f_{vz} = 44,1$  kHz je pro následnou reprodukci naprosto plnohodnotný. Chceme-li jej však upravit v

DAWu, využijeme vyšší vzorkovací frekvence. Přehled vzorkovací frekvence vzhledem k aplikaci nalezneme v tabulce 1.2; jsou to příklady z nejběžnější praxe, výjimky samozřejmě existují.

Vzorkovací frekvence	Použití
8	Telefon
32	Camcodery
44,1	CD
48	Digitální mixážní pulty
96	Nahrávací studia

Tab. 1.2: Přehled použití různých velikostí vzorkovací frekvence.

### 1.5.2 Důvody navýšení bitové hloubky

Bitová hloubka má přímý vliv na dynamiku signálu. Její nárůst v závislosti na bitové hloubce můžeme vidět v tabulce 1.3.

Bitová hloubka	Dynamika	Využití
4	25,84	Pro audio žádné
8	49,92	Telekomunikace
16	98,08	CD
24	146,24	Standard pro nahrávání
32	194,4	Zvýšení pohodlí
48	290,72	a možností úprav,
64	387,04	velká dynamika

Tab. 1.3: Závislost dynamiky signálu na bitové hloubce převodníku a typické možnosti jejího využití. Čerpáme ze zdrojů [7] a [8].

Uvedená dynamika se vztahuje k ideálním případům, v praxi se setkáváme s dynamikou mírně nižší. Důvodem je, že v praxi jsme zřídka schopni využít celou bitovou hloubku a naplnit ji signálem ať už kvůli estetice, nepředvídatelnosti dynamiky zdroje signálu nebo pochybení. Využití co nejvíce dynamického rozsahu, který je k dispozici, navíc vyžaduje kvalifikovanější obsluhu a její zvýšenou pozornost. Proto s výhodou používáme převodníky s vyšším dynamickým rozsahem, kde není nutnost jej využít celý [7, strana 40].

Dále si uvědomme, že bitová hloubka je pouze rozlišovací schopnost převodníku. Má tedy přímý vliv i na SNR (Signal-To-Noise Ratio), kdy vlastním šumem

převodníku zaplníme stále stejný počet kvantizačních hladin, ale poměr s užitečným signálem se razantně zlepší. Rovněž kvantizačního šumu vyprodukujeme při převodu méně, vzhledem k přesnějšímu vyjádření úrovně signálu ve chvíli pořízení vzorku.

Bitová hloubka převodníku má i přímý vliv na pohodlí při práci s ním. Určuje totiž rozsah dynamiky převodníku od nejnižší zaznamenané hodnoty po jeho přebuzení, tzv. clip. Z toho vyplývají dvě skutečnosti.

Zaprvé, za použití převodníku s vyšší bitovou hloubkou si můžeme dovolit předzesílit signál méně, a tedy si nechat větší rezervu, tzv. headroom, např. pro dynamické části skladby. Pokud totiž nahrajeme signál, který zaplní jen 50% z rozsahu převodníku, dostaneme se na dynamiku cca 48 dB u 16bitového převodníku proti dynamice cca 70 dB u převodníku 24bitového.

Zadruhé, jak bylo uvedeno výše, dynamika znamená nejvyšší možný rozdíl od nejnižšího zaznamenaného zvuku po přebuzení. Má tedy přímý vliv na možné hlasitostní rozdíly. Byť si u běžné komprimované popové nahrávky vystačíme s dynamikou 40 dB, u koncertu filharmonie může být dynamika 70 dB málo.

Poslední, avšak neméně důležitá výhoda, kterou přináší navýšení bitové hloubky, je příprava k postprodukčním úpravám v DAWu. Jakoukoli změnou, ať už ekvalizací, úpravou hlasitosti nebo přimícháním efektu, změníme i úroveň jednotlivých vzorků signálu, a je tedy nutné jim přiřadit novou kvantizační hladinu. Proto, pokud použijeme převodník s nízkou bitovou hloubkou, znásobí se kvantizační šum vzniklý při rekvantizaci s každou úpravou.

Proto dnešní, profesionální DAW pracují interně v režimu 32 bitů float. Znamená to, že i když je samotný projekt uživatelem nastaven na 16 bitů, resp. 24 bitů, DAW veškeré změny počítá v režimu 32 bitů float a navolenou bitovou hloubku použije až při finálním uložení na disk.

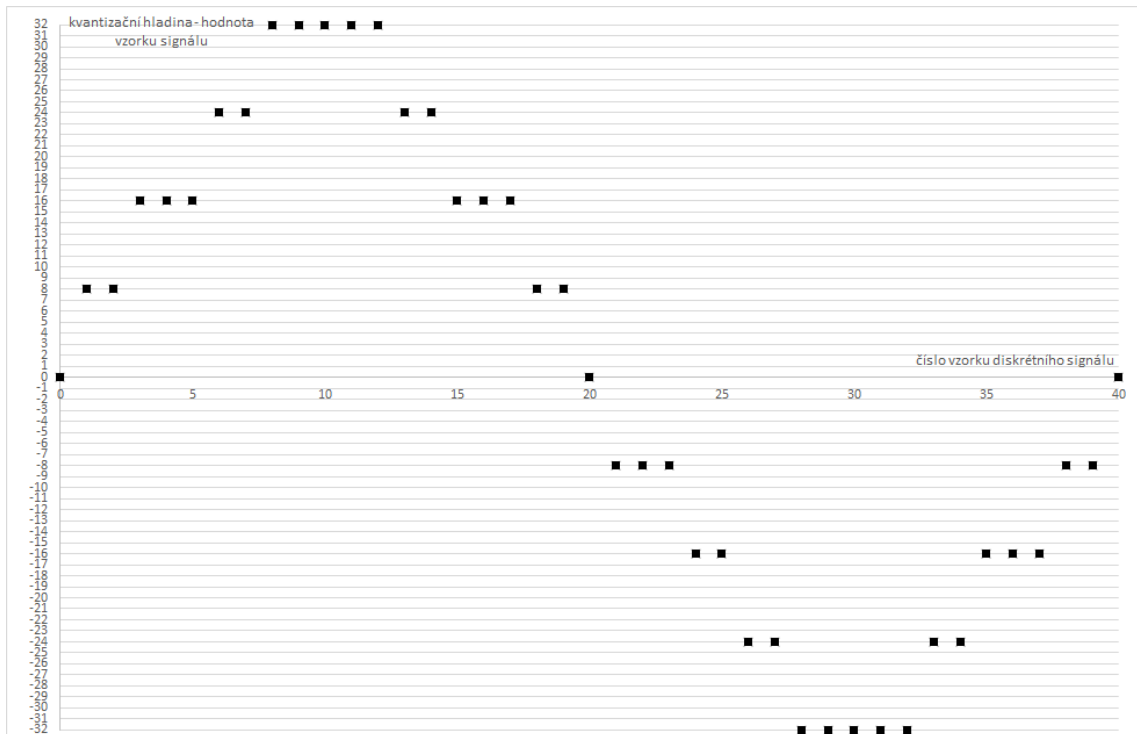
## 2 ZPŮSOB NAVÝŠENÍ BITOVÉ HLOUBKY

### 2.1 Zero padding

Při umělém navyšování bitové hloubky, tedy při dekvantizaci, nedochází k zisku nových informací, jak bychom očekávali. Již vstupní signál např. s bitovou hloubkou 16 bitů je degradovaný, neboť každý z jeho vzorků nabývá jedné hodnoty z 216, tj. 65 536 kvantizačních hladin. Tedy, i když jej dekvantizujeme na 24bitovou hloubku, získáme sice nově 224, tj. 16 777 216 hodnot jednotlivých kvantizačních hladin, ale plníme je informacemi pouze o 65 536 z nich. Velmi zjednodušeně tedy můžeme říci, že dekvantované vzorky zaplní každou 256. hladinu, ale byt přenášíme informace o hodnotách zbylých 255 hladin, žádný ze vzorků je nenabývá. Toto lze vidět na obrázku 2.1. Nedochází tedy k žádnému zvýšení kvality. Tato metoda je nazývána zero-padding. Formálně jde o pronásobení každého vzorku hodnotou dva umocněnou kladným rozdílem bitových hloubek (původní a nové):

$$Y = X \cdot 2^{(q-p)} \quad (2.1)$$

kde  $q$  představuje novou bitovou hloubku a  $p$  původní bitovou hloubku. Viz [5], [6].



Obr. 2.1: Výsledek dekvantizace signálu bez získání nové informace – nové kvantizační hladiny zůstávají volné. Konkrétní případ dekvantizace ze 3 na 5 bitů.

Tento proces má význam například pro dosažení kompatibility dvou audio materiálů při práci v postprodukci, kdy zpracováváme více materiálů v různých bitových hloubkách, ale potřebujeme docílit jednotné bitové hloubky v projektu daného DAW. Opět nedokážeme využít nově nabytý prostor pro informace a navíc v případě větších postprodukčních zásahů do nahrávky se může stát slyšitelným rozdíl materiálu s bitovou hloubkou 16 bitů v přímém srovnání s materiálem s hloubkou 24 bitů, nehledě na rozdíl dynamik nahrávek.

Druhá motivace takto uměle navýšit bitovou hloubku materiálu je právě práce v postprodukci, kdy se zásahy do nahrávky zapisují s využitím nově nabytých kvantizačních hladin, tedy zaplní se i hladiny nevyužité. Jde totiž o to, že signál, který podrobíme ekvalizaci, zesílení či efektu, může nabýt hodnoty, již předtím neobsahoval. Jednoduše tímto procesingem přidáme do reprezentace signálu nové informace, které se při zápisu na disk kvantují s novou bitovou hloubkou.

Zero-padding je totiž proces, který ke každému vzorku přistupuje samostatně bez ohledu na to, co se nachází před ním či za ním. S využitím i těchto informací o okolí vzorku se ovšem můžeme dostat k podstatně lepším, uspokojivějším výsledkům.

## 2.2 Praxe při úpravě signálu

Veškeré profesionální DAW i externí procesory pracují za použití bitové hloubky vyšší, než je bitová hloubka signálu, který zpracovávají. S výhodou zde využíváme vlastností digitálního signálu oproti analogovému, kdy nemusíme uvažovat reálnou amplitudu signálu ve voltech. Můžeme si dovolit navýšit dynamiku signálu, a zajistit si tak dostatečnou rezervu. K zvýšení úrovně zpracovávaného signálu dochází například při smíchání podobných signálů (v praxi typicky jeden zdroj snímáný více způsoby), tedy amplituda výsledného mixu bude větší. Pro zajištění dostatečné dynamické rezervy se dříve používaly analogové masteringové mixážní pulty, které na výstupu zpracovávaly signál o reálné amplitudě až 120 VDC [9, strana 98, 99].

Standardně v nahrávacích studiích zaznamenáváme signál za použití převodníků s bitovou hloubkou 24 bitů. Při úpravě však použitím např. bitové hloubky pevných 32 bitů docílíme osmibitového rozšíření dynamiky signálu, tedy 48 dB. Pokud bychom chtěli dosáhnout takovéto úrovně analogovým zařízením, zpracovávali bychom signál o amplitudě až 3 000 VDC. Za použití interní vyšší bitové hloubky dosahujeme tedy nesrovnatelně větších rezerv a možností, přičemž obsluhu nehrozí úraz při práci s vysokým napětím.

Nejběžnější praxí je však použití plovoucích 32, resp. 64 bitů. Amplitudu vstupního signálu vyrenderujeme na 25, resp. 53 bitů, čímž dostaneme 8, resp. 11 bitů proměnný exponent. Ten nám umožní při úpravách překročit pevný limit bitové

hloubky bez zkreslení. Zpracováváme tedy i úrovně signálu mimo rozsah, do kterého se musíme vrátit až před zpětným D/A převodem. Zmírníme tím i kvantizační chybu, protože reálně použitý bitový řetězec pro každý vzorek zůstává neovlivněn touto chybou, ta se objeví až v exponentu. Tím vznikne stejné množství kvantizačního šumu, ať už kvantujeme slabý nebo silný signál.

## 3 ŘÍDKÉ REPREZENTACE SIGNÁLŮ

Jak bylo výše rozebráno, běžné zvýšení bitové hloubky, tedy dekvantizace audio signálu metodou zero-padding, nabídne spoustu možností pro procesing a postprodukční úpravu nahrávky. Tato dekvantizace však nepřináší žádné nové informace do bitové reprezentace audio signálu a má smysl právě a jen pro navýšení prostoru pro následné zásahy do něj. Nepomůže nám však například, pokud byl pro záznam signálu použit A/D převodník s příliš nízkou bitovou hloubkou, pokud byl při záznamu signál málo předzesílen nebo pokud byla provedena chybná postprodukční úprava, například digitální zeslabení a následná rekvantizace signálu. Poslední uvedený případ je konkrétně rozebrán dále v této práci.

Sofistikovanější dekvantizace se snaží odpovědět na otázku, jakou hodnotu by nabyl vzorek záznamu, kdyby byl při nahrávání použit převodník s vyšší bitovou hloubkou. Pokud na tuto otázku odpovídáme zkoumáním jednoho izolovaného vzorku, nejsme schopni získat odpověď. O jednotlivém vzorku máme jen dvě informace, a to je jeho pozice v čase a amplituda reprezentovaná bitovým vyjádřením s počtem bitů odpovídajícím počtu bitů převodníku použitého při záznamu. Pokud se na něj ale podíváme v širším kontextu, víme, jaký vzorek mu předcházel, jaký po něm následuje, a také víme to, že to není obecný signál, ale audio signál, který má specifické vlastnosti. Jak bylo psáno výše, slyšitelný, tedy pro nás podstatný audio signál nabývá frekvencí od 20 Hz po 18 kHz. Dále víme, že audiosignál je spojitý a reprezentovaný součtem relativně malého počtu funkcí sinusového průběhu. Víme, že tón obsahuje kromě fundamentu vyšší harmonické složky, a umíme předpovědět jejich pozici a amplitudu. Všechny tyto vlastnosti ústí ve fakt, že jsme schopni najít báze, ve kterých je audiosignál řídký. Pokusme se tedy rozebrat možnost dekvantizace audio signálu na základě jeho řídké reprezentace.

### 3.1 Představení řídké reprezentace signálů

V následující podkapitole budeme čerpat ze zdrojů [10], [5]. Pro podrobnější informace odkazujeme na tyto zdroje.

#### 3.1.1 Značení

Pro větší přehlednost budeme níže značit skalární veličiny kurzívou, např.  $m$ ,  $N$ , vektory tučně malými písmeny, tedy:  $\mathbf{x}$ ,  $\mathbf{y}$ , a matice tučně velkými písmeny:  $\mathbf{A}$ ,  $\mathbf{B}$ . Vektory uvažujeme jako sloupcové a jejich prvky indexujeme od jedničky, tedy první prvek vektoru  $\mathbf{x}$  značíme  $x_1$ ,  $n$ -tý pak  $x_n$ .

### 3.1.2 Pojmy

**Kardinalitou množiny** rozumíme počet jejích prvků. Značíme ji podobně jako absolutní hodnotu, tedy  $|\{2, 8, -6, 0\}| = 4$ .

**Nosičem vektoru** nazýváme množinu indexů, jejichž prvky mají nenulovou hodnotu. Značíme jej  $\text{supp}(\mathbf{x})$ , tedy  $\text{supp}(\mathbf{x}) = \{i \mid x_i \neq 0\}$ . Pro signál  $\mathbf{x} = [2, 6, 0, -5, 0, 3]^T$  píšeme  $\text{supp}(\mathbf{x}) = \{2, 6, -5, 3\}$ , takže  $|\text{supp}(\mathbf{x})| = 4$ .

**Norma vektoru.** Pro další vysvětlení ještě potřebujeme definovat  $\ell_p$ -normu vektoru. Pro jakýkoli vektor  $\mathbf{x} \in \mathbf{C}^N$  je definována:

$$\begin{aligned}\|\mathbf{x}\|_p &:= \left(\sum_{i=1}^N |x_i|^p\right)^{\frac{1}{p}} \text{ pro } 1 \leq p < \infty, \\ \|\mathbf{x}\|_p &:= \sum_{i=1}^N |x_i|^p \text{ pro } 0 < p < 1, \\ \|\mathbf{x}\|_\infty &:= \max_i |x_i|, \\ \|\mathbf{x}\|_0 &:= |\text{supp}(\mathbf{x})|.\end{aligned}\tag{3.1}$$

Setkáváme se i se speciálními případy  $\ell_p$ -norm:

$\ell_1$ -norma, tedy  $\|\cdot\|_1$  – součet absolutních hodnot prvků vektoru

$\ell_0$ -norma, tedy  $\|\cdot\|_0$  – počet nenulových složek vektoru

$\ell_\infty$ -norma, tedy  $\|\cdot\|_\infty$  – vrací složku vektoru s nejvyšší hodnotou.

**Řídký** vektor je takový vektor, který obsahuje i nulové prvky. Vektor  $\mathbf{x}$  je  $k$ -řídkým, pokud platí

$$\|\mathbf{x}\|_0 \leq k.\tag{3.2}$$

Tedy je jím vektor s nejvýše  $k$  složkami nenulovými. V návaznosti definujeme pojem relativní řídkost, což je poměr počtu nenulových složek  $k$  vektoru k jeho délce  $N$ ,  $k/N$ .

## 3.2 Aditivní model signálu

Pracujeme-li se signálem za využití jeho řídkosti, vyplatí se použít jeho vyjádření pomocí součinu matice a vektoru souřadnic. Jde tedy o transformaci, matici pak nazýváme transformační. V případě Fourierovy transformace transformační matice obsahuje funkce sinusového průběhu a vektor souřadnic je soubor koeficientů pro pronásobení této matice.



Pokud tedy chceme tímto modelem zapsat signál  $\mathbf{y}$ , využijeme transformační matice  $\mathbf{A}$  a vektoru souřadnic (souboru koeficientů)  $\mathbf{x}$ . Zápis tedy vypadá:

$$\mathbf{y} = \mathbf{A} \cdot \mathbf{x}, \quad (3.3)$$

kde  $\mathbf{y}$  zastupuje signál,  $\mathbf{A}$  transformační matici a  $\mathbf{x}$  pak vektor souřadnic.

### 3.3 Řídké řešení

Víme tedy, že můžeme zpracovávaný signál zapsat jako součin matice a vektoru. Tato úloha má však spoustu řešení, ale aby tahle metoda byla funkční, musíme najít právě to řešení s největším počtem nulových složek, tedy to nejřidší. Formálně zapisujeme

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{vzhledem k} \quad \mathbf{Ax} = \mathbf{y}, \quad (3.4)$$

kde  $\min_{\mathbf{x}} \|\mathbf{x}\|_0$  reprezentuje řešení s největším počtem nulových složek. Vektor  $\mathbf{y} \in \mathbf{C}^N$  reprezentuje signál a matice  $\mathbf{A} \in \mathbf{C}^{m \times N}$  definuje lineární zobrazení. Předpokládáme případy, kdy  $m < N$ , resp.  $m \ll N$ , a  $\mathbf{A}$  je plně řádkové hodnosti. Matici  $\mathbf{A}$  nazýváme *slovník*, její sloupce pak *atomy*. Všechna nalezená  $\mathbf{x}$  pak považujeme za přípustná řešení.

V praxi je ale řešení  $\mathbf{x}$  příliš mnoho a jsou mezi nimi i řešení pro nás bezcenná, protože reálný signál je zašuměn. Proto zavádíme jistou toleranci a připouštíme i řešení, kde nehledáme řešení s co nejvíce nulovými složkami, ale řešení s co nejvíce složkami zanedbatelné velikosti. Úloha se mění takto:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{vzhledem k} \quad \|\mathbf{Ax} - \mathbf{y}\|_p \leq \delta, \quad (3.5)$$

kde většinou volíme  $p = 2$ .

Nalezení řídkého řešení není jednoduchá ani samozřejmě splnitelná úloha. Slovník  $\mathbf{A}$  musí splňovat určité podmínky a řídké řešení dokonce vůbec nemusí existovat.

K nalezení nejřidšího možného řešení můžeme využít i jiné metody. Existuje totiž číslo spark, jehož minimalizací jsme schopni nalézt nejřidší řešení rovnice. Tento postup je však co do výpočetní náročnosti srovnatelný s původním řešením úlohy, nebudeme jej tedy v práci rozebírat jako pro praxi nevyhovující. Viz [10, strana 46]

**Vzájemná koherence matice (slovníku)** je definována jako největší absolutní normalizovaný skalární součin dvou různých sloupců matice, formálně:

$$\mu(\mathbf{A}) = \max_{1 \leq j, k < N, j \neq k} \frac{|a_j a_k|}{\|\mathbf{a}_j\|_2 \cdot \|\mathbf{a}_k\|_2}, \quad (3.6)$$

kde  $a_j$ , resp.  $a_k$  označuje  $j$ -tý, resp.  $k$ -tý sloupec matice  $\mathbf{A}$ .

Pokud se vzájemná koherence matice blíží nule, pak pravá strana roste nade všechny meze. Naopak pokud se blíží jedné, blíží se jedné i pravá strana. Hledáme proto co nejméně koherentní slovníky.

$\ell_1$  – **relaxace** Jelikož  $\ell_0$  – norma není konvexní, není řešení problému za jejího využití časově dostupné. Avšak  $\ell_p$  – normy jsou konvexní, proto volíme nejbližší konvexní normu, a tedy  $\ell_1$ . Zjišťujeme, že některé výsledky dosažené za využití  $\ell_1$  normy jsou shodné s těmi, které poskytuje norma  $\ell_0$ . Bereme-li v potaz příklady z praxe, kde počítáme s faktem, že signál má v sobě obsažen šum, je těchto případů většina. Zavádíme tedy jistou toleranci  $\delta$ , která reprezentuje šum obsažený v signálu. Dostáváme se k ekvivalenci formulace původní úlohy, a to:

$$\arg \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{vzhledem k} \quad \|\mathbf{Ax} - \mathbf{y}\|_2 \leq \delta. \quad (3.7)$$

Podobnost přístupu k řešení problému za využití této normy lze vidět na ilustraci objemově se zvětšujících jednotkových koulí, viz [10, strana 51].

### 3.4 Strukturovaná řídkost

Strukturovaná řídkost přistupuje ke koeficientům v širším kontextu, nezaobírá se jimi individuálně [10, strana 57, 58], [11]. Navíc víme, že můžeme zpracovávaný signál, tedy nahrávku, vyjádřit pomocí součinu vektoru souřadnic a vybrané matice neboli slovníku. Ten je pevně daný, má určenou podobu. Hledáme vektor souřadnic, jehož pronásobením slovníku získáme původní signál. Bylo zmíněno, že s výhodou využijeme, když nepřístupujeme k zpracovávanému signálu jakožto k signálu obecnému, ale bereme v úvahu jeho vlastnosti audio signálu. Tato informace pomůže k výběru slovníku, který bude obsahovat veškeré prvky, které k reprezentaci signálu potřebujeme, a minimální množství jiných. Platí, že pokud se určitý koeficient v signálu vyskytne jednou, je vysoká pravděpodobnost, že se objeví vícekrát, popřípadě jen mírně změněný. Tedy na rozdíl od vnímání koeficientů jako jednotlivostí, jak jsme činili doteď, se snažíme je sdružovat do skupin. Dostáváme se k pojmu skupinová řídkost, kdy jsou některé skupiny koeficientů vysoce důležité a jiné naopak zanedbatelné.

Výsledkem těchto poznatků je zjištění, že je ideální mít slovník vytvořený, resp. naučený pro každou dekvantizační úlohu individuálně.

#### 3.4.1 Strukturovaná řídkost v audio signálech

Můžeme říci, že nepotřebujeme žádné informace k rekonstrukci signálů s frekvenčními složkami mimo 20 Hz – 18 kHz. Dále, pokud máme libovolný vzorek signálu,

můžeme zjistit, zda jde o ruch, hudbu či mluvené slovo, přičemž u hudby jsme schopni určit žánr a použité harmonie, v případě řeči zase rozlišit mužský a ženský hlas.

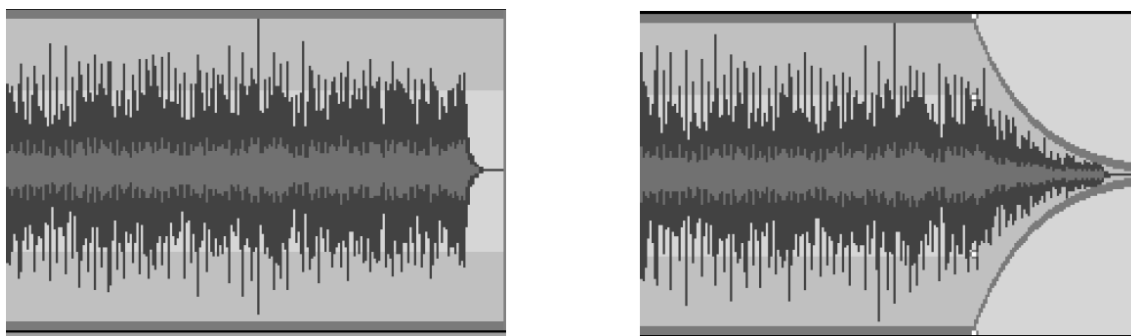
Z Gaborovy transformace audio signálu například lze vyčíst, že v transientním obsahu signálu dochází k vertikální a v harmonickém k horizontální korelaci. Dostáváme se k zadání separační úlohy:

$$\arg \min_{\mathbf{X}_{\text{TO}}, \mathbf{X}_{\text{TR}}} \frac{1}{2} \|\mathbf{A} \cdot (\mathbf{X}_{\text{TO}} + \mathbf{X}_{\text{TR}}) - \mathbf{y}\|_2^2 + \lambda_{\text{TO}} \|\mathbf{X}_{\text{TO}}\|_{2,1} + \lambda_{\text{TR}} \|\mathbf{X}_{\text{TR}}\|_{2,1}. \quad (3.8)$$

kde matice  $\mathbf{X}_{\text{TO}}$ ,  $\mathbf{X}_{\text{TR}}$  jsou hledané matice Gaborových syntetizačních koeficientů a  $\mathbf{A}$  je gaborovský slovník. Minimalizaci čtverců odchylek zajistí první člen, druhý hlídá řídkost nalezených řádkových reprezentací. Řídkost sloupcových reprezentací pak zajišťuje třetí člen. Konstantami  $\lambda_{\text{TO}}$  a  $\lambda_{\text{TR}}$  umíme ze signálu získat tonální složku syntézou  $\mathbf{A} \cdot \text{vec}(\mathbf{X}_{\text{TO}})$ , transienty pak syntézou  $\mathbf{A} \cdot \text{vec}(\mathbf{X}_{\text{TR}})$  [10, strana 60].

## 4 SIGNÁL MODIFIKOVANÝ EFEKTEM FADE-OUT, RESP. FADE-IN

Vytyčme zadanou úlohu. Pracujeme s nahrávkou modifikovanou efektem fade-out, resp. fade-in. Pro jednoduchost budeme zpočátku uvažovat jen fade-out. Tuto nahrávku chceme vrátit do podoby co nejvíce podobné její původní, tedy při co nejmenší ztrátě informace. Kromě nahrávky máme k dispozici pouze obálku použitého efektu, příklad je zobrazen na obrázku 4.1



Obr. 4.1: Aplikace efektu fade-out. Příklad zobrazuje známou obálku.

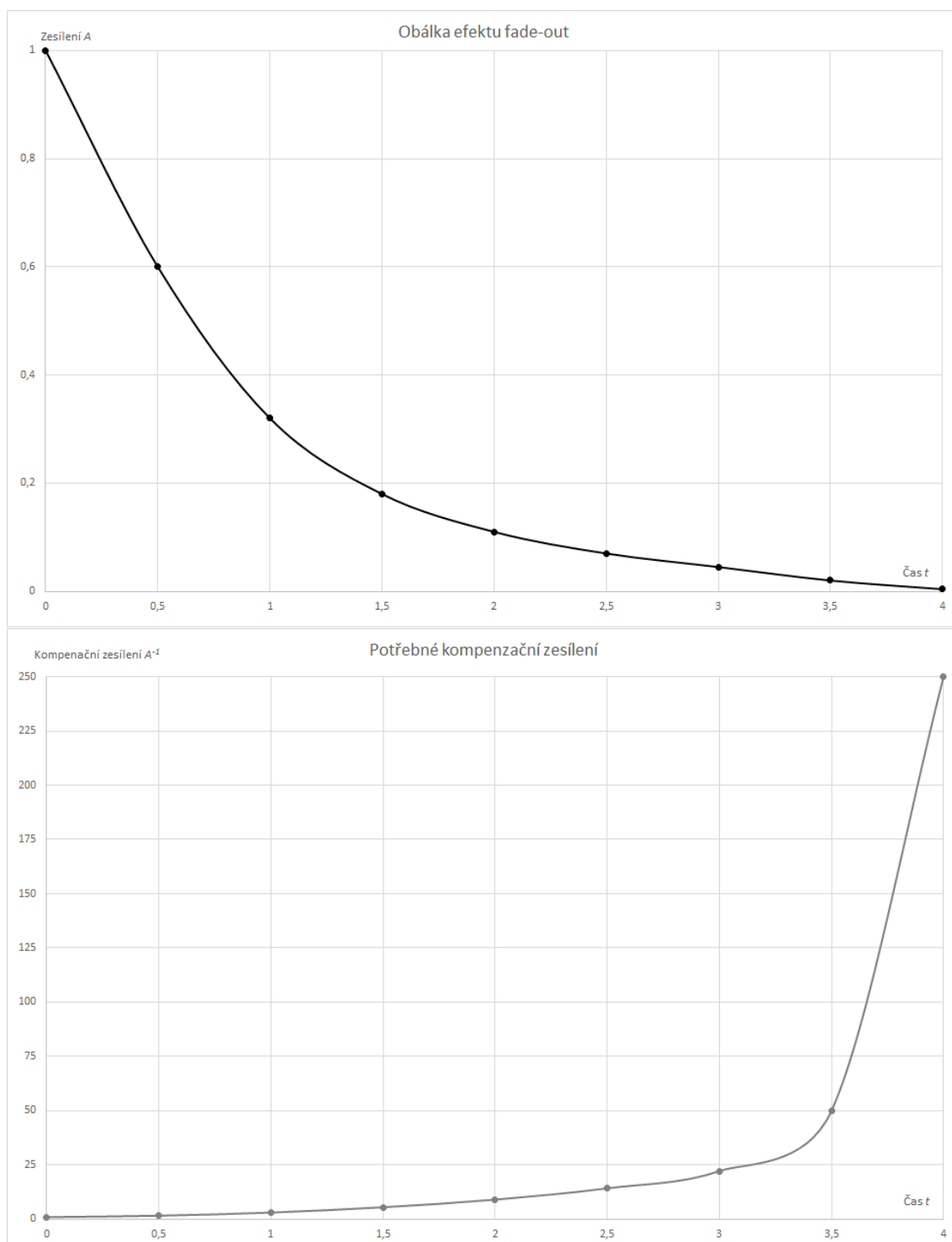
### 4.1 Simulace řešení úlohy v analogové oblasti

Pro lepší pochopení simulujme řešení úlohy v analogové oblasti. Na magnetofonovém pásku je nahrávka zeslabená do ztracena (český ekvivalent pro fade-out). Víme, jakým průběhem byla nahrávka zeslabena, a pokud se při reprodukci pokoušíme zeslabení kompenzovat, zvyšujeme potenciometrem hlasitost výstupního signálu, abychom dosáhli zesílení o průběhu inverzním. Ilustrováno na obr. 4.2.

Samozřejmě, pokud bychom se snažili ztlumení kompenzovat takto, spolu se slabším signálem zesílujeme i šum všech prvků reprodukčního řetězce a navíc, pokud se ke konci efektu dostaneme na nulu, není již co zesílit, nehledě na reálná omezení kompenzačního zesilovače (ke konci dráhy obálky bychom potřebovali jeho zesílení zvýšit k nekonečnu).

### 4.2 Standardní řešení zadání

Řešme uvedený problém v oblasti digitální. Jestliže dostaneme nahrávku postiženou efektem fade-out ve stavu po rekvantizaci, již není možnost původní informace



Obr. 4.2: Znázornění kompenzace efektu zesílení.

získat zpět přesně. Pokud totiž ztlumíme nějakou část skladby a takto ji vyexportujeme, dochází nutně k opětovné kvantizaci. Ztlumený signál nabývá jen nízkých kvantizačních úrovní, a zaplní tedy nutně menší počet z nich. Když se snažíme opětovně signál zesílit na původní hodnotu, vzorky se nemohou z principu vrátit na své

správné místo. Zkusme si řešení ověřit na následujícím příkladu. Vzorek  $x$  v čase  $s$  nabývá 16 342. hladiny 16bitového převodníku. Nabývá tedy přibližně poloviny rozsahu kladné půlvlny signálu. Tento vzorek je postižen efektem fade-out, který v čase sníží původní rozsah na 28,263% z něj. Vzorek  $x$  je tedy kvantován na hladinu 4 618,739 46, to znamená na hladinu 4 619. Při prostém vynásobení zpět vzorek nabude hodnoty 16 342,921 84, takže se zaokrouhlí na 16 343. kvantizační úroveň. Nastane chyba jedné kvantizační hladiny, která vznikne v důsledku dvojí kvantizace, poprvé dolů, podruhé nahoru.

Jediného zlepšení situace můžeme docílit, když opětovnou dekvantizaci nahoru provedeme za využití větší bitové hloubky, například 24 bitů. Pokud totiž před dekvantizací zvýšíme bitovou hloubku, ve které děláme úpravu, zvýšíme rozlišovací schopnost. Kvantizační hladina 4 619 u 16bitového převodníku odpovídá podle vzorce (2.1) 1 182 464. kvantizační hladině u převodníku 24bitového. Jestliže tuto hodnotu budeme dekvantizovat, dostaneme se na hodnotu 4 183 787,991, tedy 4 182 788. Zanedbáme tedy 0,009 místo 0,07816 kroku. Původní zanedbání je navíc ještě větší, protože kvantizační krok u 24bitového převodníku je nesrovnatelně menší než u 16bitového, takže 0,009 kroku je daleko menší chyba. U jiného vzorku můžeme dosáhnout zlepšení o celý krok nebo více. Metoda tedy přinese alespoň mírné snížení nárůstu kvantizačního šumu.

## 4.3 Řešení za využití řídkosti signálu

### 4.3.1 Formulace úlohy pro konstantní zvýšení bitové hloubky

Víme, že audio signál má reprezentace, ve kterých je řídký. V této práci budeme využívat Gaborova slovníku, který je obecně vhodný pro práci s audio signálem. Celá úloha je dosti podobná úloze, kterou ve své bakalářské práci řešil Bc. Záviška [5], a proto na něj navážeme. Oproti této práci formuloval úlohu pro dekvantizaci obrazového signálu, navíc nezahrnoval proměnnou vstupní bitovou hloubku, ale snažil se provést konstantní navýšení po celé délce signálu. Jeho úloha byla formulována takto:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{vzhledem k} \quad \|\mathbf{y}_q - \mathbf{A}\mathbf{x}\|_\infty \leq \alpha/2, \quad (4.1)$$

kde vektor  $\mathbf{y}_q$  reprezentuje nakvantovaný signál a  $\alpha$  pak vzdálenost mezi kvantizačními hladinami.  $\mathbf{A}$  je báze, ve které je náš signál řídký. Pravá polovina definice zaručí, že výsledná hodnota vzorku nebude o více než polovinu vzdálenosti mezi kvantizačními hladinami rozdílná vzhledem k předchozí, takže nedojde k chybě celého kvantizačního kroku původní bitové hloubky. Používáme zde normu  $\|\cdot\|_\infty$  vracející prvek s největší velikostí. Pokud totiž největší odchylka stále splňuje podmínku být menší

než polovina kvantizačního kroku, pak tuto podmínku splní všechny. Stejný výsledek přináší i práce Ing. Navrátilové [12].

### 4.3.2 Formulace úlohy pro signál zatížený efektem fade-out

Výše uvedený postup řešení vyhovuje, jen nerespektuje dynamickou bitovou hloubku vstupu, která vznikne efektem fade-in/fade-out. Přesto je úloha ve své podstatě obdobná. Jediný problém, který musíme vyřešit, je ten, že u každého vzorku signálu, který dekvantujeme, úlohu provádíme z jiné bitové hloubky na jeho původní. Při převedení do oblasti řídkých reprezentací to znamená, že každý vzorek nalezený reprezentací pronásobíme koeficientem aktuálního zeslabení obálky. Pro přístup ke každému vzorku signálu zvlášť ještě jednou připomeneme význam proměnných ze zápisu (3.3). Vektor  $\mathbf{y}$  je vektor původního signálu a  $\mathbf{x}$  je vektor, kterým násobíme matici  $\mathbf{A}$  pro získání  $\mathbf{y}$ , což znamená, že první prvek vektoru  $\mathbf{y}_1$  odpovídá prvnímu vzorku zpracovávaného signálu,  $n$ -tý vzorek  $\mathbf{y}_n$  pak  $n$ -tému. Obálku si můžeme při každém vzorku představit jako násobení koeficientem, který bude  $k_n \leq 1$ . V oblasti, kde efekt fade-in nebo fade-out nepůsobí, je  $k_n = 1$ , v oblasti jeho působení pak odpovídá konkrétnímu stavu obálky efektu. Pokud chceme efekt kompenzovat, musíme každý vzorek násobit koeficientem inverzním ke koeficientu, kterého nabývá v daném momentu efekt.

Finální zápis hledání nejbližšího vektoru  $\mathbf{x}$  bude podobný zápisu (4.1), ale zároveň musíme nalezené vzorky ošetřit implementací koeficientu obálky  $k$ , respektive jejich pronásobením inverzním kompenzačním koeficientem  $k^{-1}$ . Proto k původnímu zápisu reprezentace signálu s využitím řídkosti ze (3.3) zavádíme matici  $\mathbf{W}_k$ , která nese na své diagonále koeficienty inverzní ke koeficientům reprezentujícím průběh obálky. Formálně pak:

$$\mathbf{y}_k = \mathbf{W}_k \cdot \mathbf{A} \cdot \mathbf{x}, \quad (4.2)$$

kde  $\mathbf{y}_k$  značí kompenzovaný signál,  $\mathbf{x}$  vektor koeficientů a  $\mathbf{A}$  Gaborovu bázi.

Formulujme tedy úlohu. Pro tuto úlohu mějme na počátku nekonečně přesný signál. Je potřeba ji sestavit tak, aby byla schopná individuálního přístupu k jednotlivým vzorkům signálu a u každého ze vzorků zohlednila aktuální stav koeficientu obálky efektu. Dále musíme zachovat vhodně zavedenou podmínku z (4.1), aby nově nalezený vzorek zůstal na kvantizační hladině nejvýše o polovinu kvantizačního kroku vzdálené oproti předchozí. Pokud se vzorek nachází v oblasti, kde není postižen efektem, je tato podmínka splněna již v původním znění úlohy. Rozmezí hodnot, kde vzorek před postižením efektem ležel, je však přímo úměrné velikosti jeho zeslabení. Při vytyčení tohoto rozmezí musíme tedy také zohlednit koeficient

obálky. Znění úlohy po zapsání všech výše popsaných podmínek bude následující:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{vzhledem k} \quad |\mathbf{k}_n^{-1}(\mathbf{y}_q)_n - (\mathbf{A}\mathbf{x})_n|_\infty \leq \alpha/2 \cdot \mathbf{k}_n^{-1} \forall n, \quad (4.3)$$

kde  $\mathbf{x}$  je stále hledaný nejřidší vektor souřadnic,  $\mathbf{k}_k^{-1}$  je koeficient inverzní k obálce efektu pro  $n$ -tý vzorek,  $(\mathbf{A}\mathbf{x})_n$  představuje součin Gaborovy báze s nalezeným vektorem koeficientů pro konkrétní,  $n$ -tý vzorek, tedy zastupuje námi nalezenou reprezentaci, a  $\alpha/2$  znamená polovinu kvantizačního kroku, zde ovlivněnou inverzním koeficientem vůči koeficientu obálky. Úlohu řešíme pro všechna  $n$ .

## 4.4 Proximální operátor

Proximální operátor je definován jako řešení minimalizačního problému. Hledá nejmenší možné řešení. V probírané tematice jsou využívány především proximální operátory pro  $\ell_1$ -normu, což je měkké prahování. Pokud je nalezené  $\mathbf{x}$  v přípustné množině, algoritmus vrátí nulu, je tedy minimální. Jinak vrátí  $\mathbf{x}$  plus nebo minus mez. Obdobně funguje indikátorová funkce, která vrací nulu, pokud  $\mathbf{x}$  náleží zadané množině, a nekonečno, pokud nenáleží. Projekce na množinu je pak funkce, která  $\mathbf{x}$ , které náleží zadané množině ponechá nezměněno a to, které ne, posune k jejím hranicím.

## 4.5 Navržení algoritmu pro řešení úlohy

Na řešení výše uvedené úlohy lze použít řadu algoritmů, my využijeme proximálních [10, strana 75 a dále]. Z hrubého dělení můžeme jmenovat algoritmy využívající  $\ell_1$ -relaxaci, zvané relaxační, které spoléhají na získání řešení přesného nebo alespoň relativně blízkého. Proximální algoritmy nejsou nejrychlejší dostupné, ale využíváme jejich vysokou flexibilitu. Vzhledem k zavedení  $\ell_1$ -relaxace naše úloha spadá do kategorie konvexních.

### 4.5.1 Formulace úlohy do vzoru proximálního algoritmu

Zadejme si poměrně obecnou konvexní tzv. neomezenou úlohu:

$$\arg \min_{\mathbf{x}} f(\mathbf{x}) + g(\mathbf{L}\mathbf{x}), \quad (4.4)$$

kde  $\mathbf{L}$  značí lineární operátor, v našem případě matici. Do výše popsané formy musíme přepsat úlohu (4.3). Formulace bude následující:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 + \iota_{\mathbf{C}}(\mathbf{A}\mathbf{x}), \quad (4.5)$$



kde  $\|\mathbf{x}\|_1$  nahrazuje funkci  $f(\mathbf{x})$  a index  $\mathbf{C}$  zavádí znění indikátorové funkce reprezentující naši obálku signálu. Plné znění úlohy lze zapsat jako:

$$\mathbf{C} = \{\mathbf{z} \mid |\mathbf{z}_n - k_n^{-1}(\mathbf{y}_q)_n| \leq \alpha/2 \cdot k_n^{-1}\}, \quad (4.6)$$

### 4.5.2 Volba algoritmu

Při původní úvaze byl vybrán algoritmus Chambolle-Pock, s úspěchem používaný sice především v oblasti zpracovávání obrazu, přesto zdánlivě nejvhodnější i pro výše popsanou úlohu. Po přistoupení k samotné realizaci úlohy se však ukázalo, že jednodušší řešení bude zůstat u algoritmu Douglas-Rachford. Tento algoritmus byl použit i v původním programu *demo\_dequantization.m*. Původně bylo uvažováno, že po nahrazení typu transformace z diskrétně kosinové na Gaborovu (DCT nemůžeme pro audio použít, funguje jen pro simulovaný signál) nebude možno uplatnit algoritmus Douglas-Rachford, a to z důvodu, že Gaborova transformace na rozdíl od DCT nemá ortonormální bázi vektorového prostoru ([10], str.24). Situace se ale mění při zařazení podmínky, že pro Gaborovu transformaci bude použit tzv. těsný frame 4.5.3. Toho lze docílit. Tato skutečnost umožní pro minimalizační úlohu zachovat algoritmus Douglas-Rachford, byť je použita transformace, která nemá ortonormální bázi 4.5.3.

### 4.5.3 Frame, těsný frame, Gaborův frame

Při reprezentaci signálu pomocí nedoručených systémů lineárních rovnic, jako se děje v téhle úloze, využíváme algoritmy hledající reprezentaci, ve které je signál v nějakém systému určen jen několika málo prvky. Taková reprezentace se nazývá řídkou. Systém, ve kterém určujeme signál, je určen lineární kombinací generátorů. Ty působí ve vektorovém prostoru, ve kterém je reprezentován signál. Pokud počet generátorů převyší dimenzi vektorového prostoru, pak jeden vektor může mít více reprezentací - tato vlastnost se nazývá *nedoručenost*. Nyní lze zavést pojem *Báze vektorového prostoru*, což je minimální systém jeho generátorů. A konečně - frame je taková báze, která má počet generátorů vyšší, než je dimenze vektorového prostoru. Tato nadbytečnost naopak způsobuje větší flexibilitu framů oproti bázím. Framy mají horní a dolní mez. Pokud má být dosažena podmínka, aby frame byl těsný, musí se spodní mez rovnat horní. Těsný frame je rovnocenný s ortogonální bází, čehož využíváme právě v 4.12.

### 4.5.4 Algoritmus Douglas-Rachford

Algoritmus proximálního dělení. Jedná se o zobecnění dopředně-zpětného algoritmu. Ten však předpokládá jednu funkci diferencovatelnou podle Lipschitzova spojitého

gradientu [5]. V zadané úloze se takový případ nevyskytuje. Douglas-Rachfordův algoritmus tuto podmínku nahrazuje pomocí proximálního operátoru. Větší univerzálnosti je dosaženo větší výpočetní náročností. Obecný zápis Douglas-Rachfordova algoritmu pak vypadá následovně[17]:

$$\begin{aligned}
\text{Zvolíme } \quad & \epsilon \in (0, 1), \gamma > 0, \mathbf{y}_0 \in \mathbb{R}^N \\
& n = 0, 1, \dots \\
& \mathbf{x}_n = \text{prox}_{\gamma f_2} \mathbf{y}_n \\
& \lambda_n \in \langle \epsilon, 2 - \epsilon \rangle \\
& \mathbf{y}_{n+1} = \mathbf{y}_n + \lambda_n (\text{prox}_{\gamma f_1} (2\mathbf{x}_n - \mathbf{y}_n) - \mathbf{x}_n).
\end{aligned} \tag{4.7}$$

Zápis minimalizační úlohy bude však odlišný od zápisu v *demo\_dequantization.m*. Původní úloha řešená algoritmem totiž byla dle [16] formulována takto:

$$\text{proj}_{\{\mathbf{x} | \mathbf{A}\mathbf{x} \in [b_1, b_2]\}}(\mathbf{z}) = \mathbf{z} - \mathbf{A}^+ (\mathbf{A}\mathbf{z} - \text{proj}_{[b_1, b_2]}(\mathbf{A}\mathbf{z})) \tag{4.8}$$

Zde byla využita ortonormální báze u DCT, takže pokud má matice  $\mathbf{A}$  ortonormální bázi, pak platí:

$$\mathbf{A} \dots \text{ONB} \Leftrightarrow \mathbf{A}^* = \mathbf{A}^{-1} = \mathbf{A}^+, \tag{4.9}$$

kde  $\mathbf{A}^*$  je matice adjungovaná a  $\mathbf{A}^+$  je matice pseudoinverzní (viz. [10], strana 21). S využitím 4.9 lze 4.8 zapsat jako

$$\text{proj}_{\{\mathbf{x} | \mathbf{A}\mathbf{x} \in [b_1, b_2]\}}(\mathbf{z}) = \mathbf{z} - \mathbf{A}^{-1} (\mathbf{A}\mathbf{z} - \text{proj}_{[b_1, b_2]}(\mathbf{A}\mathbf{z})), \tag{4.10}$$

odkud po triviálním roznásobení vyplýne

$$\text{proj}_{\{\mathbf{x} | \mathbf{A}\mathbf{x} \in [b_1, b_2]\}}(\mathbf{z}) = \mathbf{A}^{-1} \text{proj}_{[b_1, b_2]}(\mathbf{A}\mathbf{z}). \tag{4.11}$$

Proto mohl být zápis `f2.prox` v programu *demo\_dequantization.m* zapsán jako

Výpis 4.1: Původní zápis funkce `f2.prox` v prostředí Matlab.

```
1 f2.prox = @(x,T) At(proj_box(A(x),[],param))
```

Vzhledem k tomu, že Gaborova transformace ONB není, nelze toto zjednodušení použít. Proto musela být úloha psána v původním tvaru 4.8. Při použití těsného frame však platí alespoň, že:

$$\text{násobení } \mathbf{G}^+ \text{ je totéž jako násobení } \mathbf{G}^*, \text{ což je analýza.} \tag{4.12}$$

Pokud úloha 4.8, kde  $\mathbf{A}$  značí DCT, bude upravena pro DGT, tedy přepsána s  $\mathbf{G}$ , pak platí:

$$\text{proj}_{\{\mathbf{x} | \mathbf{G}\mathbf{x} \in [b_1, b_2]\}}(\mathbf{z}) = \mathbf{z} - \mathbf{G}^+ (\mathbf{G}\mathbf{z} - \text{proj}_{[b_1, b_2]}(\mathbf{G}\mathbf{z})). \tag{4.13}$$

Což lze efektivně rozebrat.

$\mathbf{Gz}$  – syntéza  
 $\text{proj}_{[b_1, b_2]}(\mathbf{Gz})$  – projekce do mezí  
 $\text{pak } (\mathbf{Gz} - \text{proj}_{[b_1, b_2]}(\mathbf{Gz}))$  – rozdíl syntézy a projekce  
 $(\mathbf{Gz} - \text{proj}_{[b_1, b_2]}(\mathbf{Gz})) \Rightarrow \mathbf{u}$  – s využitím 4.12 je  $\mathbf{u}$  jen analýzou  $(\mathbf{Gz} - \mathbf{t})$   
 $(\mathbf{z} - \mathbf{u})$  – poslední krok 4.13.

V prostředí MatLab:

Výpis 4.2: Nový zápis funkce f2.prox v prostředí Matlab.

```

1 f2.prox = @(x,T) x - frana(F, ((postpad(frsyn(F, x),N)
2           - proj_box(postpad(frsyn(F, x),N),[],param))))

```

Lze si všimnout, že krom příkazů **frana** a **frsyn**, což je analýza a syntéza za využití Gaborovy transformace, oproti 4.1 využíváme funkci **postpad**, která ořezává, resp. prodlužuje nulami konec vektoru dle nastavené délky. Gaborova transformace na rozdíl od diskrétní kosinové totiž může vrátit vektor delší než původní signál. Vzorky navíc však zpravidla bývají zanedbatelně malé, jde jen o jakýsi „přesah“ způsobený transformací.

## 5 PROGRAMOVÉ ŘEŠENÍ - SIMULOVANÝ A REÁLNÝ SIGNÁL

### 5.1 Programové řešení - simulovaný signál

#### 5.1.1 Výchozí stav

Pro programové řešení úlohy se s výhodou vychází z kódu publikovaného na webu unlocboxu *demo\_dequantization.m* [14]. Tento kód zpracovává prostou dekvantizaci (navýšení bitové hloubky) popisovanou v [5]. Program využívá diskretní kosinovu transformaci a nezohledňuje žádnou obálku signálu, signál ovlivňuje stejně po celé délce. Navíc pojednává o obecném, řídkém signálu. Tato práce se zaměří na optimalizaci kódu pro audio signál. V následujících podkapitolách jsou rozebrány jednotlivé části původního kódu. V každé je zaznamenána úprava na nové zadání.

#### 5.1.2 Základní parametry signálu

Zde jsou popsány základní parametry simulovaného signálu. Nově jsou zde zavedeny i parametry pro známou obálku efektu.

#### 5.1.3 Slovník a vytvoření simulovaného signálu

Pomocí anonymních funkcí a diskretní kosinové transformace je zaveden slovník pro generování řídkého signálu. Dále je vytvořen náhodný, k-řídký vektor koeficientů. Pomocí těchto dvou prvků se realizuje simulace analogového, řídkého signálu. Nově je zde určen těsný frame pro Gaborovu transformaci. Využíváme toolboxu LTFAT [15]. Viz. zdroj [10]. Dále je na signál aplikována obálka efektu určena v 5.1.2.

#### 5.1.4 Kvantizace signálu

V této části programu probíhá kvantizace signálu. Zde jen oproti původnímu kódu probíhá implementace obálky do výpočtu rozhodovacích úrovní a umístění kvantizačních hladin.

#### 5.1.5 Vykreslení dosavadních výsledků

Obstarává jen kontrolu zobrazením grafů dosavadních prvků.

### 5.1.6 Dekvantizace za využití řídkosti

Zde tkví podstata celého kódu, dekvantizace signálu do jeho původního stavu. Oproti *demo\_dequantization.m* podle [14] byla přepsána funkce *f2.eval* a *f2.prox*. Funkce *f2.eval* nově pro zjednodušení vrací nejmenší nenulovou hodnotu, vzhledem k tomu, že výsledek je ošetřen projekcí stejně, jako byl dříve ošetřen zde. Funkce *f2.prox* neprovádí projekci pomocí DCT, ale za využití Gaborovy transformace pomocí [15]. Tato změna je stěžejní, proto ji popisujeme v kapitole 4.5.2, kde 4.1 je původní zápis kódu a 4.2 ukazuje nový. Pro přehlednost je vhodné porovnat rozdíly v matematickém zápisu úloh, a to 4.8 je původní úloha, a 4.13 je úloha nová. Tu zde můžeme takto použít za předpokladu, že frame bude těsný, viz [10].

### 5.1.7 Vykreslení výsledků

Tato fáze vykreslí výsledky pro přehledné ověření funkčnosti kódu.

## 5.2 Programové řešení - reálný signál

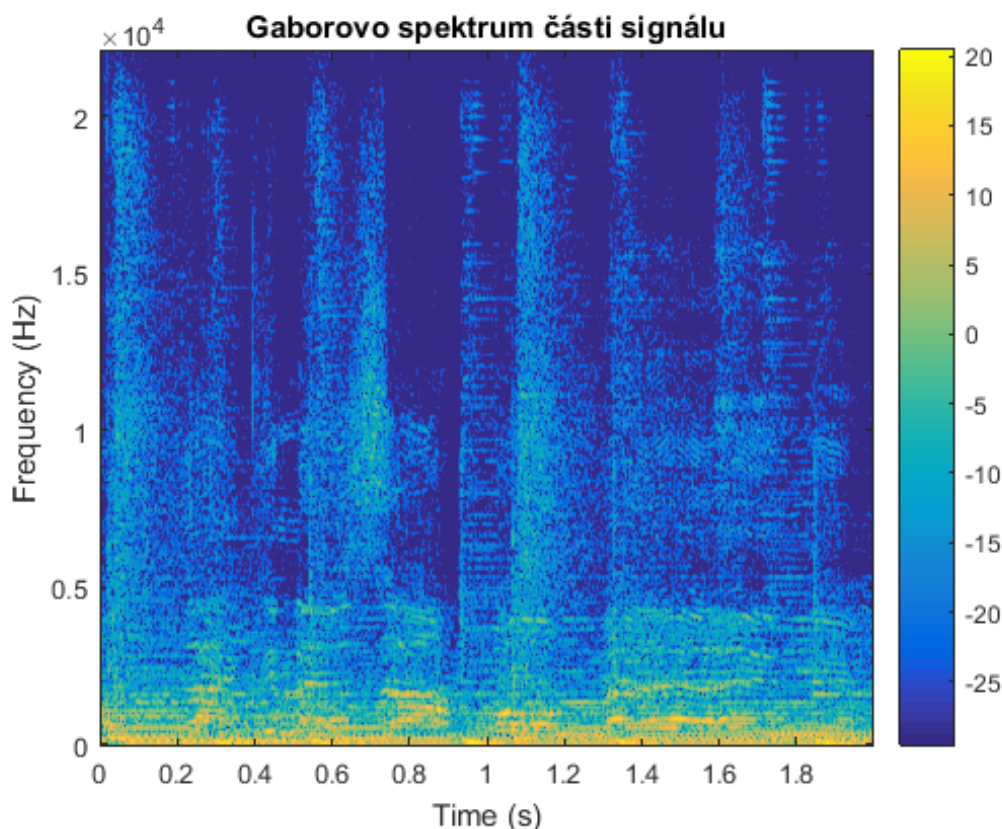
Program *dequantization\_real\_envelope.m* z velké části využívá již upravený kód *demo\_dequantization.m*. Probíhá zde nahrání reálného signálu, restrukturalizace a analýza výsledných dat. Následně budou v bodech popsány jednotlivé části programu.

### 5.2.1 Nahrání a analýza signálu, zobrazení spektra

V této části program nahrává předpřipravený mono signál řídkého audia. Připravené vzorky jsou ve formátu .wav a nahrány v 16 bitech se vzorkovací frekvencí 44.1kHz. Vzorkovací frekvenci program rozliší, ale počet bitů analyzovaného signálu je programu sdělen manuálně. Signál je analyzován (vzorkovací frekvence, délka signálu) a je vykresleno jeho spektrum. Kontrola řídkosti signálu může být provedena vizuálně, díky zobrazenému Gaborovu spektru 5.1.

### 5.2.2 Definice obálky

Zde je definován průběh obálky efektu. Tato verze programu pracuje s předpokladem, že je průběh obálky známý. To je simulováno tak, že signál je nahrán bez postižení obálkou a ta je, se zadaným průběhem, aplikována až v programu. Její tvar tedy je znám a lze s ním dále pracovat.



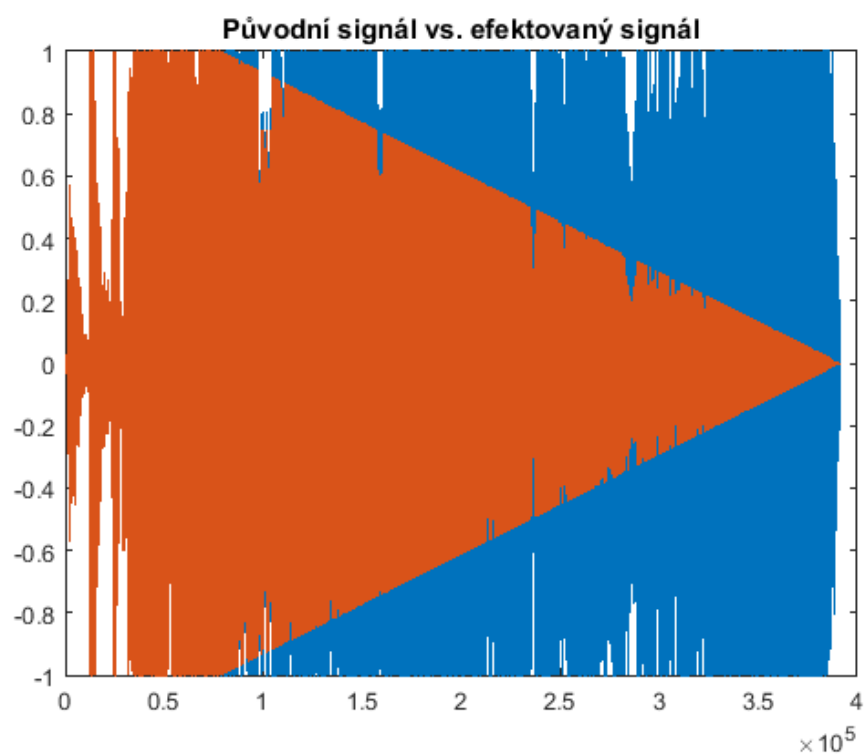
Obr. 5.1: Názorný příklad řídkého spektra testovaného signálu

### 5.2.3 Aplikace obálky

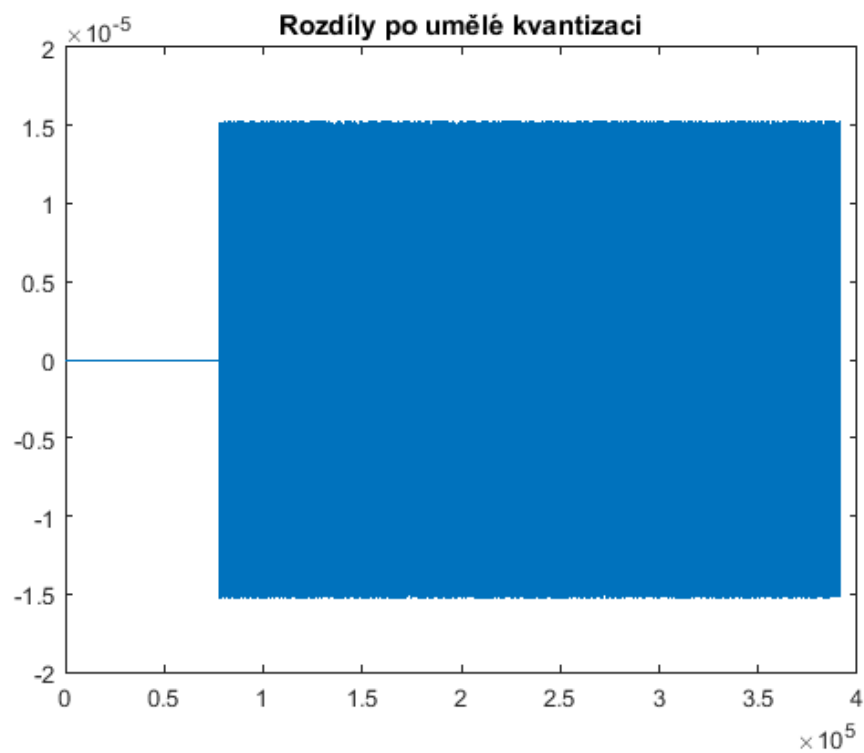
Vstupní signál je zatížen efektem fade-out. Je zobrazen graf původního a zatíženého signálu 5.2.

### 5.2.4 Určení kvantizačních hladin, rozhodovacích úrovní a kvantizace ztlumeného signálu

Vzhledem k tomu, že signál byl ztlumen pronásobením každého vzorku koeficientem obálky, jednotlivé vzorky leží mimo kvantizační úrovně původního  $n$ -bitového převodu. Je nutné poškozený signál znovu kvantizovat. V této části jsou na základě manuálně nastavené bitové hloubky určeny jednotlivé kvantizační a rozhodovací úrovně a signál je kvantizován. Program dále vykreslí rozdíl signálu před touto operací a po ní. Z grafu lze vyčíst, že chyba skokově začne na prvním vzorku, kde byla aplikována obálka, její velikost nepřekročí polovinu kvantizačního kroku 5.3. Jedná se o případ, kdy umělá kvantizace probíhá ve stejné bitové hloubce, v jaké byl kvantován vstupní signál. Proto graf před první změnou oproti originálu neukazuje žádnou chybu. Kvantizace tedy ovlivní signál dle očekávání.



Obr. 5.2: Znázornění vlivu efektu fade-out na původní signál



Obr. 5.3: Velikost změny na jednotlivých vzorcích, ke které došlo při simulaci kvantizace

### 5.2.5 Pouhé pronásobení signálu inverzní obálkou

Simulujeme rekonstrukci prostým pronásobením postiženého signálu obálkou inverzní pro porovnání výsledků s rekonstrukcí pomocí řídké dekvantizace.

### 5.2.6 Definice těsného framu

Aby mohla být použita projekce ve znění podle [16], musí být zajištěno, že frame pro Gaborovu transformaci bude těsný. Toho se docílí použitím příkazu `frame` a `frametight` podle [15]. V této části programu tedy definujeme těsný frame.

### 5.2.7 Dekvantizace za využití řídkosti - rekonstrukce efektu fade-out

Stěžejní část programu, ve které aplikujeme algoritmus. Efekt fade-out je kompenzován projekcí do množiny, jejíž hranice jsou ovlivněny průběhem opačným k průběhu obálky. Hledáme vektory koeficientů vyhovujícím našim množinám. Nalezený vektor použijeme k syntéze a získu výsledného signálu. Ten je jen zkrácen na původní délku signálu postiženého obálkou. Opět dochází k jevu, kdy z Gaborovy transformace vzejde signál mírně delší, avšak poslední vzorky (přesahující původní signál) jsou zanedbatelné velikosti. Vzhledem k faktu, že se jedná o minimalizační úlohu, byly nastaveny parametry pro ukončení iterací proximálního operátoru. Iterace skončí, pokud výsledek poslední z nich bude vyhovovat podmínce tolerance, zde nastavené na  $10^{-7}$ , nebo dokud nedovrší maxima, zde 300 iterací. Nutno poznamenat, že při dekvantizaci simulovaného signálu byla vždy splněna dříve podmínka první, naopak u reálného signálu nebyl zaznamenán případ, že by program přestal iterovat před dosažením maxima.

### 5.2.8 Analýza

U jednotlivých vzorků měříme SNR původního signálu, signálu rekonstruovaného pouhým pronásobením a signálu rekonstruovaného řídkou dekvantizací.

### 5.2.9 Vykreslení výsledků

Vykreslí výsledky, tedy rozdíl původního signálu oproti pouze pronásobenému a původního oproti rekonstruovanému.



### 5.2.10 Parametry ovlivňující výsledek dekvantizace

Na výsledek dekvantizace mají značný vliv převážně dva parametry, a to délka okna Gaborovy transformace a bitová hloubka, ve které probíhá kvantizace signálu po aplikaci efektu.

**Bitová hloubka,** ve které byl efektovaný signál exportován, je naprosto stěžejním parametrem. Prvotní testy probíhaly na jedenácti kvantizačních hladinách, což nejsou ani plné čtyři bity (šestnáct hladin). Při užití malého počtu kvantizačních hladin v jednu chvíli vzorky signálu postiženého efektem fade-out nenabývají kvantizační hladiny rozdílné od nuly. Zde není odkud zjistit koeficienty pro následnou Gaborovu transformaci. Pokud vzorky nabývají výhradně nulových hodnot, skutečně již není možnost, alespoň dle zkoumané literatury, signál zrekonstruovat. Jinými slovy, čím větší bitovou hloubkou je postižený signál kvantován, tím více z něj je možné zachránit. Při jemném rozlišení zůstává použitelná informace i na konci působení efektu. Analyzovaným parametrem výsledku dekvantizace z různých bitových hloubek je Signal-to-Noise ratio.

**Velikost okna Gaborovy transformace** má také vliv na výsledek úlohy dekvantizace. Je třeba zmínit, že na rozdíl od změny bitové hloubky délku okna Gaborovy transformace lze měnit i po obdržení „špatného“ signálu k rekonstrukci. Zvětšování Hannova okna Gaborovy transformace má za následek drobné zlepšení v rekonstrukci signálu, nicméně nelze hovořit o zásadním posunu k lepšímu. Konkrétní hodnoty nalzenete v 6.3.2

## 6 VÝSLEDKY BAKALÁŘSKÉ PRÁCE

### 6.1 Kritéria hodnocení, testovací vzorky audia

Úspěšnost procesu rekonstrukce signálu je zvažována porovnáním dvou rekonstrukcí. První byla provedena pouhým pronásobením signálu obálkou inverzní vůči obálce efektu a druhá byla provedena za pomoci řídké dekvantizace. Úspěšnost těchto metod je hodnocena podle dvou kritérií. Prvním je Signal-to-Noise ratio a druhým prostý rozdíl původního a rekonstruovaného signálu (dále jen „chybový signál“). Program je testován na čtyřech různých vzorcích. Vždy se jedná o pár sekund (5-10) relativně řídkého audio signálu.

Vzorek č.1 je vzorek populární hudby, vzorek č.2 je vzorek folku, č. 3 je lidský hlas a č. 4 pak jednoduchá kytara. Vzorky byly vybrány jakožto řídké audio signály.

### 6.2 Spektra jednotlivých vzorků

Aby celý proces fungoval, musí být rekonstruovaný signál pokud možno řídký. Spektra jednotlivých vzorků můžeme vidět na grafech 6.1 až 6.4.

### 6.3 Signal-to-Noise ratio

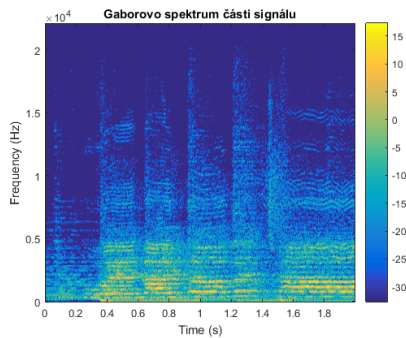
Jakožto stěžejní kritérium byl zvolen poměr signál/šum. Porovnává se poměr signálu původního oproti jednoduše pronásobenému kompenzační obálkou a původního oproti řídké dekvantovanému. Pro výpočet SNR je použit vztah (6.1).

$$20 \log_{10} \frac{\|y\|_2}{\|y - y_{\text{recon}}\|_2} \quad (6.1)$$

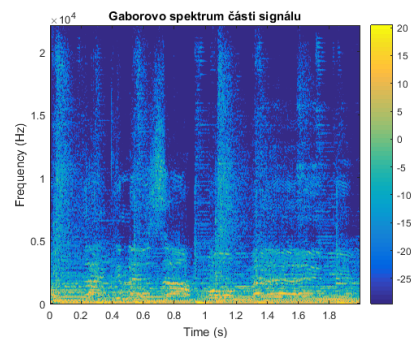
**Případ kvantizace čtyřmi bity** je uveden jen pro představu a demonstraci vlivu bitové hloubky na rekonstrukci. Čtyři bity je bitová hloubka pro zpracování audia zcela nedostačující 1.3.

#### 6.3.1 SNR při rekonstrukci z různých bitových hloubek pronásobením a řídkou dekvantizací

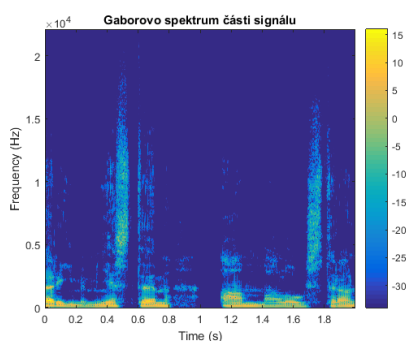
Porovnáváme dva typy rekonstrukce původního signálu kvantovaného čtyřmi, osmi, deseti a šestnácti bity.



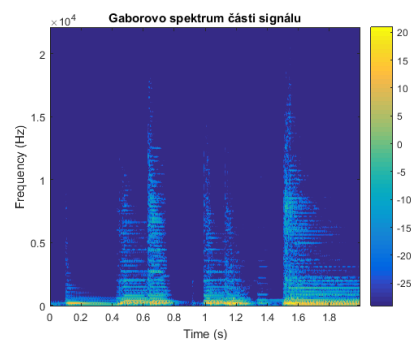
Obr. 6.1: Pop



Obr. 6.2: Folk



Obr. 6.3: Řeč

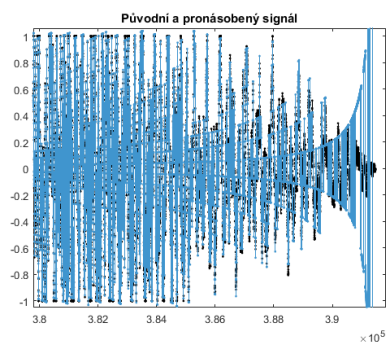


Obr. 6.4: Kytara

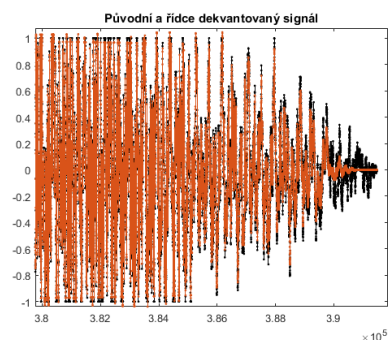
Signal-to-Noise ratio jednotlivých metod shrnujeme v tabulce 6.1. Z uvedených hodnot je jasné patrné, že čím horší vstup je ošetřován, tím lepší výsledky řídka dekvantizace oproti prostému pronásobení přináší. Pokud je vstup postižen efektem a následně kvantován šestnácti bity, je srovnatelný výstup obou metod. Pokud však je kvantován čtyřmi, osmi, nebo deseti bity, je metoda pronásobením natolik destruktivní, že samotný šum nese větší energii než původní signál a i poměr signál/šum je záporný. I po testu subjektivním poslechem dopadá podstatně lépe rekonstrukce za využití řídkosti signálu. Nejvíce se nedokonalosti rekonstrukce projeví ke konci signálu. Černý signál je původní, modrý pronásobený, oranžový řídka dekvantovaný. Zde má tendence pouhé pronásobení opisovat průběh obálky a na grafu to vytváří výrazné „paprsky“ chybných vzorků 6.5. Naopak řídka dekvantizace se zde ztrácí a „nestíhá“ sledovat původní průběh signálu. Jedná se ale o tak malý kousek závěru vzorku, že odeznění do ztracena nelze téměř registrovat, narozdíl od nepříjemného škrčení, které vytvoří pouhé pronásobení.

Vzorek	Čtyři bity		Osm bitů		Deset bitů		Šestnáct bitů	
	Řídká DEQ	Násobení	Řídká DEQ	Násobení	Řídká DEQ	Násobení	Řídká DEQ	Násobení
1	7,91	-44,41	22,19	-19,77	29,96	-7,60	55,04	53,44
2	8,74	-38,71	24,70	-14,07	33,24	-1,90	58,66	56,51
3	9,76	-42,98	27,69	-18,34	38,72	-6,17	70,79	69,67
4	7,69	-42,45	25,98	-17,81	36,23	-5,64	65,82	65,34

Tab. 6.1: Signal-to-noise ratio rekonstrukce z různých bitových hloubek pomocí řídké DEQ a prostým násobením



Obr. 6.5: Pouhé pronásobení



Obr. 6.6: Řídká dekvantizace

### 6.3.2 SNR různých délek okna u vzorků kvantovaných deseti bity

Jak již bylo výše popsáno, délka okna má téměř zanedbatelný vliv na rekonstrukci. Základní délka okna použitá i v dosavadních výpočtech je třicet vzorků. Zde bylo dosaženo  $\text{SNR} = 33,24 \text{ dB}$ . U okna šedesáti vzorků je dosaženo  $\text{SNR} = 33,38 \text{ dB}$ , u okna sto padesáti pak  $\text{SNR} = 33,49 \text{ dB}$ . U ostatních vzorků se lze setkat s obdobnou tendencí. Experimentálně bylo zjištěno, že pokud je okno příliš dlouhé, SNR se opět začne zmenšovat. Proto leží optimální délka okna nad třiceti vzorky, nicméně zlepšení jsou zanedbatelná vzhledem k náročnosti vyhledání ideální délky. V celé práci je proto použita délka okna 30 vzorků.

## 7 ZÁVĚR

První část práce stručně uvedla problematiku analogově-digitálního převodu, připomněla otázky-tematiku vzorkování a kvantování signálu. Přednesla potřeby praxe na práci s digitálním audio signálem a uvedla několik jejích příkladů.

Následně byla provedena rešerše standardních metod pro dekvantizaci audia. Ta ukázala, že krom klasické metody zero-padding, popsané v části 2.1, se další metody dekvantizace audia běžně nepoužívají, nebo nejsou známy dostupným zdrojům.

Dále byla popsána problematika řídké reprezentace signálů. Vyzdvihnuty pak byly body nezbytné pro práci s audio signálem. Jakožto inspirace pro další postup v řešení úlohy byl představen pojem strukturovaná řídkost.

Pomocí řídké reprezentace signálu byla formulována úloha k nalezení reprezentace signálu modifikovaného efektem fade-in/fade-out.

Tato formulace byla dále zkoumána a po představení proximálních algoritmů byla zapsána modifikace úlohy pro potřeby zpracování těmito algoritmy. Po přezkoumání možností zápisů úlohy, a tedy jejího řešení, byl vybrán optimální algoritmus, který byl dále rozebrán.

V další části práce byly podrobně vysvětleny dva programy v prostředí MatLab. První z programů je optimalizován pro práci se simulovaným signálem, zatímco stěžejní program práce pracuje s reálnými vzorky. Po podrobném rozebrání částí a funkcí programu jsou konzultovány parametry, které nejvíce ovlivňují výsledky dekvantizace.

Úlohu se podařilo vyřešit za využití řídkosti signálu. Nástroj funguje a je připraven na další aplikace a zkoumání. Dle výsledků této práce je řídká dekvantizace pro rekonstrukci zvukového signálu skutečně přínosná, a to zejména tam, kde jsou vstupní data v doopravdy špatném stavu. Velikou slabinou metody řídké dekvantizace je však její výpočetní náročnost a rozhodně není v silách dnešních strojů ji provádět v reálném čase.

# LITERATURA

- [1] SMÉKAL, Zdeněk. *Analýza signálů a soustav: skripta k předmětu*. Brno: Vysoké učení technická v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2012
- [2] MIŠUREC, Jiří. *Číslíkové zpracování signálů: skripta k předmětu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2011.
- [3] MADISSETTI, V. *Digital signal processing fundamentals. 2nd ed.* Boca Raton FL: CRC Press, c2010. ISBN 9781420046069.
- [4] TWEAK. *16 Bit vs. 24 Bit Audio: internetový článek*. [online] <<http://tweakheadz.com/16-bit-vs-24-bit-audio/>>. Citováno 6. 12. 2016.
- [5] ZÁVIŠKA, Pavel. *Metody pro zvýšení bitové hloubky fotografií: bakalářská práce*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2015. 82 s. Vedoucí práce byl Mgr. Pavel Rajmic, Ph.D.
- [6] PRESONUS. *Digital Audio Basics: Sample Rate and Bit Depth*. [online] <<http://www.presonus.com/news/articles/sample-rate-and-bit-depth>>. Citováno 6. 12. 2016.
- [7] SCHIMMEL, Jiří. *Elektroakustika: skripta k předmětu*. Brno: Vysoké učení technická v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2016. ISBN 978-80-214-4716-5
- [8] SCHIMMEL, Jiří. *Studiová technika: skripta k předmětu*. Brno: Vysoké učení technická v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2015. ISBN 978-80-214-4452-2
- [9] MCCARTHY, Bob. *Sound systems: design and optimization : modern techniques and tools for sound system design and alignment. Third edition*. New York: Focal Press, 2013. ISBN 9780240521565.
- [10] RAJMIC, Pavel. *Řídké a nízkohodnostní reprezentace signálů s aplikacemi: Sparse and low-rank representations of signals with applications: zkrácená verze habilitační práce*. Brno: VUTIU, 2015. ISBN 978-80-214-5195-7.
- [11] MACH, Václav. *Doplňování chybějících vzorků v audio signálu: zkrácená verze habilitační práce*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky

- a komunikačních technologií, Ústav telekomunikací, 2016. ISBN 80-214- ISSN 1213-4198.
- [12] NAVRÁTILOVÁ, Barbora. *Aplikace řídkých reprezentací dat: diplomová práce*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2011.
  - [13] CHAMBOLLE, Antonin a POCK, Thomas. *A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging*. *Journal of Mathematical Imaging and Vision* [online]. Boston: Springer US, 1105, 40(1), 120-145 [cit. 2016-12-14]. DOI: 10.1007/s10851-010-0251-1. ISSN 0924-9907.
  - [14] PERRAUDIN, Nathanaël. *UNLocBoX - Matlab convex optimization toolbox* - [online]. <<https://lts2.epfl.ch/unlocbox/>>. Citováno 2. 3. 2017.
  - [15] PRŮŠA, Zdeněk a spol. *LTFAT The Large Time-Frequency Analysis Toolbox - All your frame are belong to us* - [online]. <<http://ltfat.sourceforge.net/>>. Citováno 2. 4. 2017.
  - [16] RAJMIC, Pavel a VESELÝ, Vítězslav. *Projection onto a multidimensional interval with a particular class of full-rank linear operators* [online]. <[http://www.utko.feec.vutbr.cz/~rajmic/papers/projection\\_frame\\_interval.pdf](http://www.utko.feec.vutbr.cz/~rajmic/papers/projection_frame_interval.pdf)>. Citováno 7. 5. 2017.
  - [17] COMBETTERS, Patrick Louis a PESQUET, Jean-Christophe. *Proximal Splitting Methods in Signal Processing* [online]. <<https://arxiv.org/abs/0912.3522>>. Citováno 29. 5. 2017.



## 8 OBSAH PŘÍLOŽENÉHO CD

K této práci je přiloženo přenosné médium CD. Nachází se na něm plné znění programu *dequantization\_real\_envelope.m*, vytvořeného v programu MatLab, verze 2015b. Dále se zde nacházejí vyexportované audio soubory ve formátu wave, sloužící k demonstraci výstupů programu. Soubory jsou rozřazeny do složek a v každé složce se nachází rovněž dokument *info.txt* s informacemi o umístěných souborech.

```
/.....kořenový adresář přiloženého CD
├── 01_matlab.....program v prostředí MatLab
├── 02_cely_proces.....demonstrace průběhu programu
│   ├── 01_Původní.wav
│   ├── 02_Fade_out.wav
│   ├── 03_Pronásobeno.wav
│   ├── 04_Dekvantováno.wav
│   └── info.txt
├── 03_folk_ruzne_bity.....rekonstrukce z různých bitových hloubek
│   ├── 01_čtyři_bity_pronásobeno.wav
│   ├── 02_čtyři_bity_dekvantováno.wav
│   ├── 03_osm_bitů_pronásobeno.wav
│   ├── 04_osm_bitů_dekvantováno.wav
│   ├── 05_deset_bitů_pronásobeno.wav
│   ├── 06_deset_bitů_dekvantováno.wav
│   ├── 07_šestnáct_bitů_pronásobeno.wav
│   ├── 08_šestnáct_bitů_dekvantováno.wav
│   └── info.txt
├── 04_deset_bitu_ruzne_vzorky . různé vzorky - původní a rekonst. z 10 bitů
│   ├── 01_abba_orig_.wav
│   ├── 02_abba_dequant_.wav
│   ├── 03_humenecko_orig_.wav
│   ├── 04_humenecko_dequant_.wav
│   ├── 05_speech_orig_.wav
│   ├── 06_speech_dequant_.wav
│   ├── 07_guit_orig_.wav
│   ├── 08_guit_dequant_.wav
│   └── info.txt
└── 174440_BP_Baca.pdf ..... tato bakalářská práce
```